International Journal on Advances in Systems and Measurements

2009 vol. 2 nr. 2&3

The *International Journal on Advances in Systems and Measurements* is Published by IARIA.
ISSN: 1942-261x
journals site: http://www.iariajournals.org
contact: petre@iaria.org

Sponsored by IARIA
www.iaria.org

International Journal on Advances in Systems and Measurements
Volume 2, Numbers 2&3, 2009

- Igor Podebrad, Commerzbank, Germany
- Leon Reznik, Rochester Institute of Technology, USA
- Chi Zhang, Juniper Networks, USA

Sensor Networks

- Steven Corroy, University of Aachen, Germany
- Mario Freire, University of Beira Interior, Portugal / IEEE Computer Society - Portugal Chapter
- Jianlin Guo, Mitsubishi Electric Research Laboratories America, USA
- Zhen Liu, Nokia Research – Palo Alto, USA
- Winston KG Seah, Institute for Infocomm Research (Member of A*STAR), Singapore
- Radosveta Sokkulu, Ege University - Izmir, Turkey
- Athanasios Vasilakos, University of Western Macedonia, Greece

Electronics

- Kenneth Blair Kent, University of New Brunswick, Canada
- Josu Etxaniz Maranon, Euskal Herriko Unibertsitatea/Universidad del Pais Vasco, Spain
- Mark Brian Josephs, London South Bank University, UK
- Michael Hubner, Universitaet Karlsruhe (TH), Germany
- Nor K. Noordin, Universiti Putra Malaysia, Malaysia
- Arnaldo Oliveira, Universidade de Aveiro, Portugal
- Candid Reig, University of Valencia, Spain
- Sofiene Tahar, Concordia University, Canada
- Felix Toran, European Space Agency/Centre Spatial de Toulouse, France
- Yousaf Zafar, Gwangju Institute of Science and Technology (GIST), Republic of Korea
- David Zammit-Mangion, University of Malta-Msida, Malta

Testing and Validation

- Cecilia Metra, DEIS-ARCES-University of Bologna, Italy
- Krzysztof Rogoz, Motorola, Poland
- Rajarajan Senguttuvan, Texas Instruments, USA
- Sergio Soares, Federal University of Pernambuco, Brazil
- Alin Stefanescu, SAP Research, Germany
- Massimo Tivoli, Universita degli Studi dell'Aquila, Italy

Simulations

- Tejas R. Gandhi, Virtua Health-Marlton, USA
- Ken Hawick, Massey University - Albany, New Zealand
- Robert de Souza, The Logistics Institute - Asia Pacific, Singapore
- Michael J. North, Argonne National Laboratory, USA

Additional reviews by:

- Carlos Alexandre Barros Mello, Universidade de Pernambuco, Brazil

International Journal on Advances in Systems and Measurements
Volume 2, Numbers 2&3, 2009

Foreword

This common issue containing volume 2/2009 and volume 3/2009 of the International Journal on Advances in Systems and Measurements published by IARIA brings together eleven papers, covering different topics in the field.

In the first paper, Louis Marchildon outlines the fact that "quantum mechanics needs interpretation" and presents different approaches to understanding how quantum mechanics answer various foundational questions. The authors of the second paper, Farid Bourennani et al., propose a new weighting measure for numerical data type, which is used for effective heterogeneous data preprocessing and classification by unified vectorization. Bernd Resch et al. present in the third paper a real-time geo-awareness approach based on an open sensing infrastructure for monitoring applications. The fourth paper, written by Raimund K. Ege et al., describes a scalable peer-to-peer-based content delivery model, paired with an access control model that balances trust in end users with a risk analysis to the data provider. P. Meumeu Yomsi et al. address in the fifth paper the problem of correctly dimensioning real-time embedded systems scheduled with fixed priority scheduling. In the sixth paper, Hans-Joachim Klein and Christian Mennerich describe a graph-based method for searching and ranking clusters of polyhedra in large crystallographic databases. R. E. Kooij et al. describe in the seventh paper a subjective experiment regarding the Quality of Experience (QoE) of channel zapping and propose a system for optimal zapping experience. The authors of the eighth paper, Mussa Bshara and Leo Van Biesen, address the problem of using fingerprinting-based positioning to locate users in WiMAX networks depending on SCORE measurements. Mario Zechner and Michael Granitzer present in the ninth paper an optimized k-means implementation on the graphics processing unit. In the tenth paper, Răzvan Deaconescu et al. present a BitTorrent performance evaluation infrastructure in order test and compare current real world BitTorrent implementations and to simulate complex BitTorrent swarms. Maciej Piechowiak et al. propose in the eleventh paper a novel multicast routing algorithm without constraints and introduce the group members arrangement as a new parameter for analyzing multicast routing algorithms finding multicast trees.

We should remember that the authors of these papers were awarded by IARIA for their original papers presented at different IARIA conferences and invited to submit extended versions for this journal. These new versions of the papers were further analyzed by our reviewers, and we should thank them for their enthusiastic and volunteer support. Since all the IARIA journals provide open-access to their papers, publishing in such a journal is an important opportunity for the researchers to disseminate their work to a wide scientific community.

Constantin Paleologu, Editor-in-Chief

International Journal on Advances in Systems and Measurements
Volume 2, Numbers 2&3, 2009

CONTENTS

Quantum Mechanics Needs Interpretation

Louis Marchildon
Département de physique, Université du Québec
Trois-Rivières, QC, Canada, G9A 5H7
louis.marchildon@uqtr.ca

Abstract—**Since the beginning, quantum mechanics has raised major foundational and interpretative problems. Foundational research has been an important factor in the development of quantum cryptography, quantum information theory and, perhaps one day, practical quantum computers. Many believe that, in turn, quantum information theory has bearing on foundational research. This is largely related to the so-called epistemic view of quantum states, which maintains that the state vector represents information on a system and which has led to the suggestion that quantum theory needs no interpretation. I will argue that this and related approaches fail to take into consideration two different explanatory functions of quantum mechanics, that of accounting for classically unexplainable correlations between classical phenomena and that of explaining the microscopic structure of classical objects. The epistemic view provides no answer to what constitutes the main question of interpretation: How can the world be for quantum mechanics to be true? I will then review three different approaches to understanding quantum mechanics, namely, Bohmian mechanics, Everett's relative states, and Cramer's transactional interpretation. I will show that these approaches answer the above question, as well as other foundational ones. This paper is written from the perspective that different logically consistent interpretations, far from leading to confusion, in fact contribute to increased understanding of the theory.**

I. Introduction

Although the answer was intended to be clear, this paper's title was formulated interrogatively in my contribution to the ICQNM 2009 conference [1]. Explicit consideration of several interpretative schemes now motivates the positive formulation.

Ever since it was proposed more than 80 years ago, quantum mechanics has raised great challenges both in foundations and in applications.[1] The latter have been developed at a very rapid pace, opening up new vistas in most branches of physics as well as in much of chemistry and engineering. Substantial progress and important discoveries have also been made in foundations, though at a much slower rate. The measurement problem, long-distance correlations, and the meaning of the wave function are three of the foundational problems on which there has been and still is lively debate.

It is fair to say that foundational studies have largely contributed to the burgeoning of quantum information theory, one of the most active areas of development of quantum mechanics in the past 25 years. Quantum information is dependent on entanglement, whose significance was brought to light through the Einstein-Podolsky-Rosen (EPR) argument [6]. The realization that transfer protocols based on quantum entanglement may be absolutely secure has opened new windows in the field of cryptography [7]. And the development of quantum algorithms thought to be exponentially faster than their best classical counterparts has drawn great interest in the construction of quantum computers [8]. These face up extraordinary challenges on the experimental side. But attempts to build them are likely to throw much light on the fundamental process of decoherence and perhaps on the limits of quantum mechanics itself [9], [10].

Along with quantum information theory came also a reemphasis of the view that the wave function (or state vector, or density operator) properly represents knowledge, or information [11], [12], [13]. This is often called the *epistemic view* of quantum states. On what the wave function is knowledge of, proponents of the epistemic view do not necessarily agree. The variant most relevant to the present discussion is that rather than referring to objective properties of microscopic objects (such as electrons, photons, etc.), the wave function encapsulates probabilities of results of eventual macroscopic measurements. The Hilbert space formalism of quantum mechanics is taken as complete, and its objects in no need of a realistic interpretation. Additional constructs, like value assignments [14], Bohmian trajectories [15], multiple worlds [16], or transactions [17] are viewed as superfluous at best.

Just like foundational studies have contributed to the development of quantum information theory, many investigators think that the latter can help in solving the foundational and interpretative problems of quantum mechanics. A number of proponents of the epistemic view believe that it considerably attenuates, or even completely solves, the problems of quantum measurement, of long-distance correlations, and of the meaning of the wave function. These problems will be summarized briefly in Sec. II, and the way the epistemic view deals with them will be presented in Sec. III. I will then argue, in Sec. IV, that the epistemic view and related approaches fail to take into consideration that quantum mechanics has two very different explanatory functions: that of accounting for classically unexplainable correlations between classical phenomena, and that of explaining the microscopic structure of classical objects [18], [19]. In Sec. V, I will ask the question of what it means to interpret quantum mechanics, or any scientific theory for that matter. Drawing from the so-called semantic view of theories, I will argue than interpreting quantum mechanics means answering the question, "How can the world be for quantum mechanics to be true?" [20].

[1] Relevant reviews and paper collections are, for instance, [2], [3], [4], [5].

The next three sections will examine how three interpretative schemes of quantum mechanics, namely, Bohmian mechanics, Everett's many worlds, and Cramer's transactional interpretation, answer the above question and attempt to solve the foundational problems. Concluding remarks will be made in the last section.

II. Three problems in quantum mechanics

Although the way to apply quantum mechanics to practical situations was never a matter of dispute, the meaning of the formalism has been problematic from the outset. The first problem concerned the ψ function that appears in Schrödinger's fundamental equation. Is it something like the electric and magnetic fields we are familiar with? Schrödinger first proposed that the absolute square of ψ is proportional to the electron's charge distribution [21]. But this was quickly found untenable. Born then proposed his probabilistic interpretation, according to which the absolute square of ψ represents the probability to find the electron at a given place. This much, an instance of what is now known as Born's rule, is universally accepted. But it is still a matter of debate whether ψ represents an individual system or a statistical ensemble of systems [22], and whether it is a real field or has a strictly operational significance.

The second problem also arose very early in the development of quantum mechanics, and concerns the question of measurement. Broadly speaking, the problem is the following. Suppose we want to describe, in a completely quantum-mechanical way, the process of measuring a physical quantity Q pertaining to a microscopic system. For simplicity, assume that the spectrum of Q is discrete and nondegenerate, that $\mathbf{x}$ stands for the coordinates of the microscopic system, and that the normalized eigenfunction $\phi_i(\mathbf{x})$ corresponds to the eigenvalue q_i. For the process to be fully described by quantum mechanics, the measurement apparatus should also be considered as a quantum system, which comes to interact with the microscopic system. Let $\alpha_0(\xi)$ denote the initial wave function of the apparatus. Here ξ stands for the one-dimensional *pointer coordinate* of the apparatus. The myriad of other apparatus coordinates, representing all its microscopic degrees of freedom, are not explicitly represented.

The interaction between the microscopic quantum system and the apparatus will represent a faithful measurement of Q if the combined system evolves like

$$\phi_i(\mathbf{x})\alpha_0(\xi) \to \phi_i(\mathbf{x})\alpha_i(\xi), \tag{1}$$

where $\alpha_i(\xi)$ represents a state of the apparatus wherein the pointer shows the value α_i (with $\alpha_i \neq \alpha_j$ if $i \neq j$).[2]

It is instructive to see how the evolution (1) can be realized explicitly. Let the interaction between the microscopic system and the apparatus take place in the interval $0 < t < T$. In that time interval, take the Hamiltonian as

$$H = gQP_\xi, \tag{2}$$

where g is a real constant and P_ξ is the momentum operator conjugate to the pointer's position operator. We have neglected, here, terms in the Hamiltonian specifically connected with the microscopic system or the pointer (for instance, $\mathbf{P}\cdot\mathbf{P}/2m$), which is a good approximation if g is sufficiently large and T is sufficiently small. If the initial combined wave function is given by $\phi_i(\mathbf{x})\alpha_0(\xi)$, the final wave function will be obtained straightforwardly [23] as

$$\begin{aligned}
\phi_i(\mathbf{x})\alpha_0(\xi) &\to \exp\left\{-\frac{iT}{\hbar}H\right\}\phi_i(\mathbf{x})\alpha_0(\xi)\\
&= \exp\left\{-\frac{iT}{\hbar}gQP_\xi\right\}\phi_i(\mathbf{x})\alpha_0(\xi)\\
&= \phi_i(\mathbf{x})\exp\left\{-\frac{iT}{\hbar}gq_iP_\xi\right\}\alpha_0(\xi)\\
&= \phi_i(\mathbf{x})\alpha_0(\xi - gTq_i)\\
&= \phi_i(\mathbf{x})\alpha_i(\xi).
\end{aligned} \tag{3}$$

In its final state α_i, the pointer is moved by a distance gTq_i from its initial state. We assume that the initial wave packet $\alpha_0(\xi)$ is sufficiently narrow for all the $\alpha_i(\xi)$ to be essentially non-overlapping.

If the Schrödinger equation is universally valid, the combined evolution of the microscopic system and macroscopic apparatus is unitary (assuming, unrealistically, that they form together a closed system). But then, an initial state involving the superposition of several eigenstates of an observable of the microscopic system evolves into a final state involving a superposition of macroscopically distinct states of the apparatus (or of the apparatus and environment in more realistic situations). Explicitly,

$$\left\{\sum_i c_i\phi_i(\mathbf{x})\right\}\alpha_0(\xi) \to \sum_i c_i\phi_i(\mathbf{x})\alpha_i(\xi). \tag{4}$$

Obviously, we never see a macroscopic apparatus in a superposition of states corresponding to different pointer readings. The discrepancy between this observation and the unitary evolution expressed in (4) constitutes the measurement problem.

To solve the problem, von Neumann suggested a long time ago that the unitary evolution breaks down somewhere in the measurement process [24]. Specifically, von Neumann postulated that in measurement interactions like (4), the right-hand side abruptly collapses into one of its components. This process is fundamentally indeterministic, and the probability that the superposition collapses into $\phi_j(\mathbf{x})\alpha_j(\xi)$ is taken to be equal to $|c_j|^2$. Von Neumann did not propose any specific mechanism accounting for the collapse of the wave function, but interesting suggestions along these lines were made in subsequent years [25].[3]

A third problem that quantum mechanics has to deal with is the one of long-distance correlations [13], [29]. Consider the realization of the EPR setup in terms of two spin 1/2 particles

[2] For simplicity, we will always assume that the system's states $\phi_i(\mathbf{x})$ don't change in a measurement.

[3] Decoherence theory no doubt helps in making the measurement problem sharper, but the present author shares the view that it is by itself insufficient to solve the problem. For recent perspectives see [26], [27], [28].

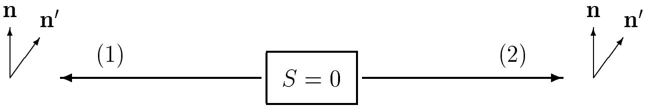


Fig. 1. Two particles prepared in the singlet state and leaving in opposite directions.

(labelled 1 and 2), depicted in Fig. 1. The state vector $|\chi\rangle$ of the compound system is taken to be an eigenstate of the total spin operator with eigenvalue zero. In this case

$$|\chi\rangle = \frac{1}{\sqrt{2}}\left\{|+;\mathbf{n}\rangle|-;\mathbf{n}\rangle - |-;\mathbf{n}\rangle|+;\mathbf{n}\rangle\right\}. \qquad (5)$$

Here the first vector in a (tensor) product refers to particle 1 and the second vector to particle 2. The vector $|+;\mathbf{n}\rangle$, for instance, stands for an eigenvector of the $\mathbf{n}$-component of the particle's spin operator, with eigenvalue $+1$ (in units of $\hbar/2$). The unit vector $\mathbf{n}$ can point in any direction, a freedom which corresponds to the rotational symmetry of $|\chi\rangle$.

Suppose Alice measures the $\mathbf{n}$-component of the spin of particle 1 and obtains the value $+1$. Then she can predict with certainty that if Bob measures the same component of the spin of particle 2, he will obtain the value -1. It then seems that the state of particle 2 changes immediately upon Alice's obtaining her result, and this no matter how far apart Alice and Bob are. Since the word "immediately", when referring to spatially separated events, is not a relativistically invariant concept, such a mechanism seems to imply instantaneous action at a distance, and is certainly not easy to reconcile with the theory of special relativity.

The interpretation of the wave function, the measurement of a quantum observable, and long-distance correlations are problems that an interpretation of quantum mechanics should clarify.

III. The epistemic and related views

In the epistemic view of quantum states, the wave function represents knowledge, or information. Let us examine the arguments that advocates of the epistemic view offer to solve the foundational and interpretative problems of quantum mechanics. I should point out that they do not all attribute the same strength and generality to these arguments. Some advocates believe that the problems are completely solved by the epistemic view, while others are of the opinion that they are just attenuated. This distinction, however, is not crucial to our purpose, and I will simply give the arguments as they are typically formulated.

The problem that is directly addressed by the epistemic view is the one of the interpretation of the wave function (or state vector, or density operator). Just as the name suggests, the state vector is normally interpreted as representing the state of quantum systems. As we have seen, some believe that the state pertains to an individual system, others to a statistical ensemble of systems. But the epistemic view, which goes back at least to writings of Heisenberg [30], claims that it represents neither. It denies that the (in this context utterly misnamed) state vector represents the state of a microscopic system. Rather, it represents knowledge about the probabilities of results of measurements performed in a given context with a macroscopic apparatus, in other words, information about "the potential consequences of our experimental interventions into nature" [13]. This is often set in the framework of a Bayesian approach, where probability is interpreted in a subjective way.

Now how does the epistemic view deal with the measurement problem? It does so by construing the collapse of the wave function not as a physical process, but as a change of knowledge [31]. Insofar as the wave function is interpreted as objectively describing the state of a physical system, its abrupt change in a measurement implies a similar change in the system, which calls for explanation. If, on the other hand, and in line with a Bayesian view, the wave function describes knowledge of conditional probabilities (i.e., probabilities of future macroscopic events conditional on past macroscopic events), then as long as what is conditionalized upon remains the same, the wave function evolves unitarily. It collapses when the knowledge base changes (this is Bayesian updating), thereby simply reflecting the change in the conditions being held fixed in the specification of probabilities.

The epistemic view also offers an explanation of long-distance correlations like the ones produced in EPR setups. We recall that when Alice obtains the value $+1$ when she measures the $\mathbf{n}$-component of her spin, she can predict with certainty that Bob will obtain -1 when he measures the $\mathbf{n}$-component of his spin. But according to the epistemic view, what changes when Alice performs a measurement is Alice's knowledge. Bob's knowledge will change either if he himself performs a measurement, or if Alice sends him the result of her measurement by conventional means. Hence no information is transmitted instantaneously, and there is no physical collapse on an equal time or spacelike hypersurface.

Related to the epistemic view is the idea of *genuine fortuitousness* [32], [33], a radically instrumentalist view of quantum mechanics. The idea "implies that the basic event, a click in a counter, comes without any cause and thus as a discontinuity in spacetime" [33, p. 405]. Indeed

> [i]t is a hallmark of the theory based on genuine fortuitousness that it does not admit physical variables. It is, therefore, of a novel kind that does not deal with things (objects in space), or measurements, and may be referred to as the theory of no things. (p. 410)

Such approaches to the interpretation of quantum mechanics are to be contrasted with realist views that we will examine later.[4]

[4] The "correlations without correlara" view of quantum mechanics [34], also known as the Ithaca Interpretation, shares with the epistemic view the idea that no reality is attributed to individual properties of quantum systems. However, correlations do have physical reality and the Ithaca interpretation strives to eliminate knowledge from the foundations.

IV. Two explanatory functions

To examine how appropriate the epistemic and related views of quantum mechanics are, it is important to properly understand the explanatory role of quantum mechanics as a physical theory. Although all measurements are made by means of macroscopic apparatus, quantum mechanics is used, as an explanatory theory, in two different ways: it is meant to explain (i) nonclassical correlations between macroscopic objects and (ii) the small-scale structure of macroscopic objects [18], [19]. That these two functions are distinct is best shown by contrasting the world in which we live with a hypothetical, closely related one [20].

Roughly speaking, the hypothetical world is defined so that (a) for all practical purposes, all macroscopic experiments give results that coincide with what we find in the real world, and (b) its microscopic structure, if applicable, is different from the one of the real world. Let us spell this out in more detail.

In the hypothetical world large scale objects, i.e., objects much larger than atomic sizes, behave just like large scale objects in the real world. The trajectories of baseballs and airplanes can be computed accurately by means of classical mechanics with the use of a uniform downward force, air friction, and an appropriate propelling force. Waveguides and antennas obey Maxwell's equations. Steam engines and heat pumps work according to the laws of classical thermodynamics. The motion of planets, comets, and asteroids is well described by Newton's laws of gravitation and of motion, slightly corrected by the equations of general relativity.

Close to atomic scales, however, these laws may no longer hold. Except for one restriction soon to be spelled out, I shall not be specific about the changes that macroscopic laws may or may not undergo in the microscopic realm. Matter, for instance, could either be continuous down to the smallest scales, or made of a small number of constituent particles like our atoms. The laws of particles and fields could be the same at all scales, or else they could undergo significant changes as we probed smaller and smaller distances.

In the hypothetical world one can perform experiments with pieces of equipment like Young's two-slit setup, Stern-Gerlach devices, or Mach-Zehnder interferometers. Let us focus on the Young type experiment. It makes use of two macroscopic objects which we label E and D. These symbols could stand for "emittor" and "detector" if it were not that, as we shall see, they may not emit or detect anything. At any rate, E and D both have on and off states and work in the following way. Whenever D is suitably oriented with respect to E (say, roughly along the x axis) and both are in the on state, D clicks in a more or less random way. The average time interval between clicks depends on the distance r between D and E, and falls roughly as $1/r^2$. The clicking stops if, as shown in Fig. 2, a shield of a suitable material is placed perpendicularly to the x axis, between D and E.

If holes are pierced through the shield, however, the clicking resumes. In particular, with two small holes of appropriate size and separation, differences in the clicking rate are observed for small transverse displacements of D behind the shield. A plot of the clicking rate against D's transverse coordinate displays maxima and minima just as in a wave interference pattern. No such maxima and minima are observed, however, if just one hole is open or if both holes are opened alternately.

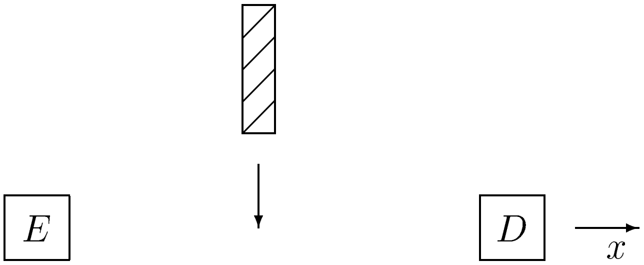


Fig. 2. Shielding material prevents D from clicking

At this stage everything happens as if E emitted some kind of particles and D detected them, and the particles behaved according to the rules of quantum mechanics. Nevertheless, we shall nor commit ourselves to the existence or nonexistence of these particles, except on one count. Such particles, if they exist, are not in any way related to hypothetical constituents of the material making up D, E, or the shield, or of any macroscopic object whatsoever. Whatever the microscopic structure of macroscopic objects is, it has nothing to do with what is responsible for the correlations between D and E.

In a similar way, we can perform in the hypothetical world experiments with Stern-Gerlach devices, Mach-Zehnder interferometers, or other setups used in the typical quantum-mechanical investigations carried out in the real world. Correlations are observed between initial states of "emittors" and final states of "detectors" which are unexplainable by classical mechanics but follow the rules of quantum mechanics. We assume again that, if these correlations have something to do with the emission and absorption of particles, these are in no way related to eventual microscopic constituents of the macroscopic devices.

In the experiments just described that relate to the hypothetical world, quantum mechanics correctly predicts the correlations between D and E (or other "emittors" and "absorbers") when suitable experimental configurations are set up. In these situations, the theory can be interpreted in (at least) two broadly different ways. In the first one, the theory is understood as applying to genuine microscopic objects, emitted by E and detected by D. Perhaps these objects follow Bohmian-like trajectories (see Sec. VI), or behave between E and D in some other way compatible with quantum mechanics. In the other interpretation, there are no microscopic objects whatsoever going from E to D. There may be something like an action at a distance. At any rate the theory is in that case interpreted instrumentally, for the purpose of quantitatively accounting for correlations in the stochastic behavior of E and D.

In the hypothetical world we are considering, I believe that both interpretations are logically consistent and adequate. Of course, each investigator can find more satisfaction in one interpretation than in the other. The epistemic view of quantum mechanics corresponds to the instrumentalist interpretation. It simply rejects the existence of microscopic objects that have no other use than the one of predicting observed correlations between macroscopic objects.

In the world in which we live, however, the situation is crucially different. The electrons, neutrons, photons, and other particles that diffract or interfere are the same that one appeals to in order to explain the structure of macroscopic objects. Denying their existence, as is done in the approach of genuine fortuitousness, dissolves such explanatory power. Denying that they have states, as is done in the epistemic view, leaves one to explain the state of a macroscopic object on the basis of entities that have no state.

V. Interpreting quantum mechanics

The epistemic and related views therefore fail to account for the second explanatory role of quantum mechanics. To reinforce this conclusion, it is instructive to investigate what it means to interpret a theory.

With most physical theories, interpretation is rather straightforward. But this should not blind us to the fact that even very familiar theories can in general be interpreted in more than one way. A simple example is classical mechanics.

Classical mechanics is based on a well-defined mathematical structure. This consists of constants m_i, functions $\mathbf{x}_i(t)$, and vector fields $\mathbf{F}_i$ (understood as masses, positions, and forces), together with the system of second-order differential equations $\mathbf{F}_i = m_i\mathbf{a}_i$. A specific realization of this structure consists in a system of ten point masses interacting through the $1/r^2$ gravitational force. A hypothesis may then assert that the solar system corresponds to this realization, if the sun and nine planets are considered pointlike and all other objects neglected. Predictions made on the basis of this model correspond rather well with reality. But obviously the model can be made much more sophisticated, taking into account for instance the shape of the sun and planets, the planets' satellites, interplanetary matter, and so on.

Now what does the theory have to say about how a world of interacting masses is really like? It turns out that such a world can be viewed in (at least) two empirically equivalent but conceptually very different ways. The first one consists in asserting that the world is made only of small (or extended) masses that interact by instantaneous action at a distance. The second way asserts that the masses produce everywhere in space a gravitational field, which then locally exerts forces on the masses. These two ways constitute two different interpretations of the theory. Each one expresses a possible way of making the theory true (assuming empirical adequacy). Whether the world is such that masses instantaneously interact at a distance in a vacuum, or a genuine gravitational field is produced throughout space, the theory can be held as truly realized.

Similar remarks apply to classical electromagnetism. The mathematical equations can be interpreted as referring to charges and currents interacting locally through the mediation of electric and magnetic fields. Alternatively, they can be viewed as referring to charges and currents only, interacting by means of (delayed) action at a distance [35].

In this respect, quantum mechanics seems different from all other physical theories. There appears to be no straightforward way to visualize, so to speak, the behavior of microscopic objects. This was vividly pointed out by Feynman [36, p. 129] who, after a discussion of Young's two-slit experiment with electrons, concluded that "it is safe to say that no one understands quantum mechanics. [...] Nobody knows how it can be like that." But the process of interpreting quantum mechanics lies precisely in taking up Feynman's challenge. It is to answer the question, "How can the world be for quantum mechanics to be true?"

If we adopt this point of view (known as the semantic view of theories [37], [38]), we can understand the motivation to look for interpretative schemes of quantum mechanics. Each such scheme provides one clear way that the microscopic objects can behave so as to reproduce the quantum-mechanical rules and, therefore, the observable behavior of macroscopic objects. In the following sections, we shall look at three such approaches, and see how each ones deals with the three problems outlined before.

VI. Bohmian mechanics

A. One particle

Bohmian mechanics [15], [39], also known as the de Broglie-Bohm theory owing to de Broglie's early work [40], is a realistic causal theory that (in its standard form) exactly reproduces the statistical results of quantum mechanics. To see how it works, consider a particle of mass m whose Hamiltonian is given by

$$H = \frac{1}{2m}\mathbf{P}\cdot\mathbf{P} + V(\mathbf{X},t). \tag{6}$$

The Schrödinger equation can be written as

$$i\hbar\frac{\partial}{\partial t}\psi(\mathbf{x},t) = -\frac{\hbar^2}{2m}\nabla^2\psi(\mathbf{x},t) + V(\mathbf{x},t)\psi(\mathbf{x},t), \tag{7}$$

where the wave function $\psi(\mathbf{x},t)$ is assumed to be normalized.

The complex function ψ can be written in polar form as

$$\psi(\mathbf{x},t) = R(\mathbf{x},t)\exp\left\{\frac{i}{\hbar}S(\mathbf{x},t)\right\}, \tag{8}$$

where R and S are two real functions. Substituting (8) in (7) and manipulating, one easily finds that

$$\frac{\partial}{\partial t}(R^2) + \boldsymbol{\nabla}\cdot\left\{R^2\frac{1}{m}\boldsymbol{\nabla}S\right\} = 0, \tag{9}$$

$$\frac{\partial S}{\partial t} + \frac{1}{2m}\,(\boldsymbol{\nabla}S)\cdot(\boldsymbol{\nabla}S) + V(\mathbf{x},t) - \frac{\hbar^2}{2m}\frac{1}{R}\nabla^2 R = 0. \tag{10}$$

Suppose that, for an initial value $\psi(\mathbf{x}, t_0)$ of the wave function, the solution of (7) has been found. Define

$$V_{\mathrm{Q}}(\mathbf{x}, t) = -\frac{\hbar^2}{2m}\frac{1}{R}\nabla^2 R \tag{11}$$

and

$$V_{\mathrm{tot}}(\mathbf{x}, t) = V(\mathbf{x}, t) + V_{\mathrm{Q}}(\mathbf{x}, t). \tag{12}$$

Equation (10) then becomes

$$\frac{\partial S}{\partial t} + \frac{1}{2m}(\boldsymbol{\nabla} S)\cdot(\boldsymbol{\nabla} S) + V_{\mathrm{tot}}(\mathbf{x}, t) = 0. \tag{13}$$

Formally, (13) coincides with the Hamilton-Jacobi equation associated with a classical particle with momentum

$$\mathbf{p} = \boldsymbol{\nabla} S \tag{14}$$

and Hamiltonian

$$H(\mathbf{x}, \mathbf{p}, t) = \frac{1}{2m}\mathbf{p}\cdot\mathbf{p} + V_{\mathrm{tot}}(\mathbf{x}, t). \tag{15}$$

That observation is at the root of the de Broglie-Bohm theory, which rests on the following hypotheses:

1) At every instant t, a quantum particle has a well-defined position $\mathbf{x}$ and momentum $\mathbf{p}$.
2) The particle's trajectory is governed by the Hamilton-Jacobi equation (13) or, equivalently, by Newton's equation

$$m\frac{d^2\mathbf{x}}{dt^2} = -\boldsymbol{\nabla} V_{\mathrm{tot}}(\mathbf{x}, t). \tag{16}$$

The potential V_{tot} is the sum of an external potential V and of the *quantum potential* V_{Q}, determined by the solution of Schrödinger's equation (7).
3) The particle's position and momentum, although well-defined, cannot be known exactly. One can only know the probability density that at time t, the particle is at point $\mathbf{x}$. This probability density is equal to $|\psi(\mathbf{x}, t)|^2 = R^2(\mathbf{x}, t)$, where $R(\mathbf{x}, t)$ satisfies (9).[5]

Since it is usually not known, and is unpredictable on the basis of the wave function alone, the well-defined particle's position, in Bohmian mechanics, is often called a *hidden variable*.

We should note that hypothesis (3) on the probability density is consistent with the trajectories of the particles that make up the statistical ensemble. Indeed let a large number of identical particles be distributed according to a density $R^2(\mathbf{x}, t)$, with velocities equal to $\frac{1}{m}\mathbf{p}(\mathbf{x}, t)$. The particle number conservation law can then be written as

$$\frac{\partial}{\partial t}(R^2) + \boldsymbol{\nabla}\cdot\left\{R^2\frac{1}{m}\mathbf{p}\right\} = 0 \tag{17}$$

which, owing to (14), coincides with (9).

One can also check the consistency of (14) and (16). Indeed taking the total time derivative of the former and making use of (10), we get

$$\begin{aligned}\frac{d\mathbf{p}}{dt} &= \frac{\partial}{\partial t}\boldsymbol{\nabla} S + (\mathbf{v}\cdot\boldsymbol{\nabla})(\boldsymbol{\nabla} S)\\ &= \frac{\partial}{\partial t}\boldsymbol{\nabla} S + \frac{1}{m}(\boldsymbol{\nabla} S\cdot\boldsymbol{\nabla})(\boldsymbol{\nabla} S)\\ &= \boldsymbol{\nabla}\left\{\frac{\partial}{\partial t}S + \frac{1}{2m}(\boldsymbol{\nabla} S)^2\right\}\\ &= -\boldsymbol{\nabla}\left\{V(\mathbf{x}, t) - \frac{\hbar^2}{2m}\frac{1}{R}\nabla^2 R\right\}.\end{aligned} \tag{18}$$

Hypothesis (3) provides the answer that Bohmian mechanics gives to the first problem raised in Sec. II, the one of the interpretation of the wave function. Here the absolute square of the wave function quantifies the best knowledge one can have of the particle's precise position. But note the difference with the epistemic view. In Bohmian mechanics, the particle always has a precise position, which we do not know exactly. In the epistemic view, no precise position is attached to the particle, and in fact there may even be no particle at all. The absolute square of the wave function, in the epistemic view, only represents the probability that, after a suitable macroscopic preparation procedure, a position-measuring macroscopic apparatus will yield such and such values. As we will see, Bohmian mechanics also correctly predicts probabilities of measurement results. But it does so, in position measurements, because the measurement result corresponds to a position the particle really has.

With hypothesis (3), Bohmian mechanics always reproduces the statistical results of quantum mechanics. It is instructive to see how this works in the paradigmatic example of the two-slit experiment. Here we look for the probability of detection at various points on a screen behind the slits. It is well known that in quantum mechanics, the detection probability when both slits are open is not the sum of the detection probability when the first slit is open and the detection probability when the second slit is open. This is often interpreted by saying that when both slits are open, one cannot affirm that the particle went through one specific slit at the exclusion of the other.

In Bohmian mechanics, however, whether one or two slits are open, any given particle goes through only one slit. How can this reproduce the interference pattern at the screen? It turns out that, for a given initial value of the wave function (say, when a particle is emitted), the solution of the Schrödinger equation behind the slits, when only one slit is open, is different from the corresponding solution when both slits are open. The quantum potential, given in (11), is therefore also different. Thus for a given value of the particle's position, its trajectory when one slit is open is different from its trajectory when both slits are open, even if in both cases the particle goes through the same slit. Bohmian trajectories in the two-slit experiment were numerically calculated by Philippidis *et al* [41]. The statistical results they obtained precisely reproduce Young's interference pattern, thereby providing an illuminating answer to Feynman's challenge.

[5] More general forms of Bohmian mechanics relax the identification of the probability density with $|\psi|^2$ [39].

B. Two particles

Bohmian mechanics can be generalized to any number of particles but, for our purposes, it will be enough to consider only two. To be explicit, we will consider in this subsection the case of two spinless particles interacting through a potential $V(\mathbf{x}_1, \mathbf{x}_2, t)$. The system's configuration space has six dimensions.

The Schrödinger equation can be written as

$$i\hbar\frac{\partial\Psi}{\partial t} = -\frac{\hbar^2}{2m_1}\nabla_1^2\Psi - \frac{\hbar^2}{2m_2}\nabla_2^2\Psi + V\Psi, \tag{19}$$

where ∇_i^2 $(i = 1, 2)$ stands for the Laplacian with respect to coordinates $\mathbf{x}_i$. Letting

$$\Psi(\mathbf{x}_1, \mathbf{x}_2, t) = R(\mathbf{x}_1, \mathbf{x}_2, t)\exp\left\{\frac{i}{\hbar}S(\mathbf{x}_1, \mathbf{x}_2, t)\right\}, \tag{20}$$

one finds two coupled equations for the real functions R and S. The equation that generalizes (10) is the Hamilton-Jacobi equation for two particles with momenta

$$\mathbf{p}_i = \boldsymbol{\nabla}_i S. \tag{21}$$

The particles interact through the potential $V_{\text{tot}} = V + V_{\text{Q}}$, where now

$$V_{\text{Q}}(\mathbf{x}_1, \mathbf{x}_2, t) = -\frac{\hbar^2}{2m_1 R}\nabla_1^2 R - \frac{\hbar^2}{2m_2 R}\nabla_2^2 R. \tag{22}$$

They follow well-defined trajectories governed by (21) or, equivalently,

$$m_i\frac{d^2\mathbf{x}_i}{dt^2} = -\boldsymbol{\nabla}_i V_{\text{tot}}(\mathbf{x}_1, \mathbf{x}_2, t). \tag{23}$$

The probability density that, at time t, the first particle is at $\mathbf{x}_1$ and the second particle is at $\mathbf{x}_2$ is given by $R^2(\mathbf{x}_1, \mathbf{x}_2, t)$. The equation that generalizes (9) represents the conservation of probability, and it ensures that the evolution of the probability density due to the particles' motion is consistent with the evolution of the wave function.

The most general solution of the two-particle Schrödinger equation (19) can be written as a sum of products of functions of the form

$$\Psi(\mathbf{x}_1, \mathbf{x}_2, t) = \sum_i \phi_i(\mathbf{x}_1, t)\alpha_i(\mathbf{x}_2, t). \tag{24}$$

Let us assume that, for some interval of time, the potential V is the sum of two terms that each involve the coordinates of one particle only, that is,

$$V(\mathbf{x}_1, \mathbf{x}_2, t) = V_1(\mathbf{x}_1, t) + V_2(\mathbf{x}_2, t). \tag{25}$$

Let us further assume that, at some time t_0 in that interval, the wave function Ψ is a product state, which means that there is only one term in the right-hand side of (24). In other words,

$$\Psi(\mathbf{x}_1, \mathbf{x}_2, t_0) = \phi(\mathbf{x}_1, t_0)\alpha(\mathbf{x}_2, t_0). \tag{26}$$

It is then easy to show that the wave function remains a product state for as long as V satisfies (25), with ϕ and α satisfying one-particle Schrödinger equations associated with potentials V_1 and V_2, respectively.

If we write

$$\phi = R_\phi\exp(iS_\phi/\hbar) \quad \text{and} \quad \alpha = R_\alpha\exp(iS_\alpha/\hbar), \tag{27}$$

one immediately sees that

$$R(\mathbf{x}_1, \mathbf{x}_2, t) = R_\phi(\mathbf{x}_1, t)R_\alpha(\mathbf{x}_2, t) \tag{28}$$

and

$$S(\mathbf{x}_1, \mathbf{x}_2, t) = S_\phi(\mathbf{x}_1, t) + S_\alpha(\mathbf{x}_2, t). \tag{29}$$

Equation (28) implies that the quantum potential V_{Q} in (22), and therefore the total potentiel V_{tot}, are sums of one-particle terms. The first particle's Bohmian trajectory is therefore independent of the second particle's coordinates, and vice versa. This conclusion also follows from (21) and (29).

In the general case where V is not the sum of one-particle terms, however, or even when it is such a sum but the initial wave function is not a product state, then the wave function at time t is given by (24) with the right-hand side having more than one term. In this case, (28) and (29) do not hold and the first particle's Bohmian trajectory will in general depend on the second particle's coordinates. This, we will see, is what accounts for the long-distance correlations.

C. The measurement problem

In a measurement interaction, the initial state of the quantum system and apparatus is a product state, which transforms into an entangled state according to (4). The problem consists in understanding why is the joint system described by only one term of the superposition.

In Bohmian mechanics, the particle whose observable is to be measured and the apparatus pointer both have well-defined positions at $t = 0$, before the measurement interaction begins. As the interaction unfolds, they follow trajectories governed by (21), to end up again at well-defined positions at $t = T$. The position of the pointer at that time is directly interpreted as the apparatus reading, which is entirely well-defined.

We can calculate the probability $P(i)$ that, at time T, the pointer shows the value α_i. This is obtained through the marginal distribution of the pointer observable, equal to the average over the particle's coordinates of the absolute square of the total wave function at T. Making use of the orthogonality of the eigenfunctions $\phi_j(\mathbf{x})$, we find that

$$\begin{aligned}\int d\mathbf{x}\,|\Psi(\mathbf{x}, \xi)|^2 &= \sum_{i,j} c_i^* c_j \alpha_i^*(\xi)\alpha_j(\xi)\int d\mathbf{x}\phi_i^*(\mathbf{x})\phi_j(\mathbf{x}) \\ &= \sum_j |c_j|^2|\alpha_j(\xi)|^2. \end{aligned} \tag{30}$$

Since the pointer's wave functions $\alpha_j(\xi, T)$ are essentially non-overlapping, the probability $P(i)$ that the pointer's position is within the support of α_i is equal to $|c_i|^2$, which is the statement of Born's rule.[6]

[6] If the measurement interaction does not yield orthogonal particle states, a similar argument can be made using the orthogonality of the final states of the environment.

When the measurement interaction is over, the pointer's wave functions α_j remain orthogonal for as long as one may care. This, in fact, is due to the myriad of degrees of freedom of the pointer other than ξ, whose evolution is different corresponding to different pointer positions. If the pointer has entered wave packet α_i, therefore, it will never end up in a different α_j. To do so, it would have to go through a region of configuration space associated with zero probability. The Bohmian trajectories of both the particle and apparatus henceforth develop as though only the i^{th} term was present. The other ones still are, but they have no effect whatsoever on subsequent trajectories. Although the wave function has never collapsed, the system evolves as if it had.

D. Long-distance correlations

As I summarized in [42], one can incorporate spin in Bohmian mechanics by adding spinor indices to the wave function, in such a way that $\Psi \to \Psi_{i_1 i_2}$. There can be several ways to associate particle spin vectors with the wave function [39], but one way or other they involve the expressions

$$\mathbf{s}_1 = \frac{\hbar}{2\Psi^\dagger\Psi}\Psi^\dagger\boldsymbol{\sigma}_1\Psi, \qquad \mathbf{s}_2 = \frac{\hbar}{2\Psi^\dagger\Psi}\Psi^\dagger\boldsymbol{\sigma}_2\Psi. \tag{31}$$

Here $\boldsymbol{\sigma}_1$ and $\boldsymbol{\sigma}_2$ are Pauli spin matrices for the two particles.

In the singlet state, the initial wave function typically has the form

$$\Psi = \phi_1(\mathbf{x}_1)\phi_2(\mathbf{x}_2)|\chi\rangle, \tag{32}$$

where $|\chi\rangle$ is given in (5). With such a wave function, it is easy to show that $\mathbf{s}_1 = 0$ and $\mathbf{s}_2 = 0$. That is, both particles initially have spin zero. This underscores the fact that in Bohmian mechanics, values of observables outside a measurement context do not in general coincide with eigenvalues of associated operators.

Spin measurement was analyzed in detail by Dewdney *et al.* [43], [44]. In the EPR context, in particular, these investigators first wrote down the two-particle Pauli equation adapted to the situation shown in Fig. 1. With Gaussian initial wave packets ϕ_1 and ϕ_2, the equation can be solved under suitable approximations. Bohmian trajectories can then be obtained by solving (21). These equations of motion involve the various components of the two-particle wave function in a rather complicated way, and must be treated numerically.

Suppose that the magnetic field in the spin-measuring apparatus on the left of Fig. 1 is oriented in the $\mathbf{n}$ direction. Consider the case where particle 1 enters that apparatus much before particle 2 enters the one on the right-hand side. What was shown was the following. When particle 1 enters the apparatus along a specific Bohmian trajectory, the various forces implicit in (21) affect both the trajectory and the spin vector, the latter building up through interaction with the magnetic field. The beam in which particle 1 eventually ends up depends on its initial position. If particle 1 ends up in the upper beam of the spin-measuring apparatus, its spin becomes aligned with $\mathbf{n}$. Meanwhile there is an instantaneous action on particle 2, simultaneously aligning its spin in the $-\mathbf{n}$ direction. Similarly, if particle 1's initial position is such that it ends up in the lower beam, its spin becomes aligned with $-\mathbf{n}$, and the spin of particle 2 simultaneously aligns in the $\mathbf{n}$ direction.

Thus the nonlocal forces, present in Bohmian mechanics as a consequence of the nonfactorizability of the wave function, have, once the measurement of the spin of particle 1 has been completed, resulted in particle 2 having a spin exactly opposed. A subsequent measurement of the spin component of particle 2 along $\mathbf{n}$ then reveals the perfect correlation predicted by quantum mechanics.

VII. Everett's relative states

Everett's relative states, or many-worlds, interpretation is an attempt to meet the challenge of interpretation while eschewing the introduction of the collapse of the wave function or of hidden variables. The wave function is taken to apply to individual systems and is meant to represent the true state of the quantum system at all times. Everett also claimed to be able to deduce Born's rule from the other postulates of quantum mechanics. That claim has been the subject of much controversy [45], but its analysis falls outside the scope of the problems raised here.

Everett considers the wave function of a compound system after a quantum measurement, represented for instance by the right-hand side of (4). Confronted with the fact that all pointer readings appear, Everett takes the bull by the horns and claims that they indeed all exist. They all exist, but each reading (say α_j) is associated with only one value of the quantum observable (in this case q_j). Everett calls $\phi_j(\mathbf{x})$ and $\alpha_j(\xi)$ *relative states*. In other words, the value q_j exists relative to α_j, and vice versa.

Since all pointer readings exist at once, understanding that multiplicity is a crucial question, to which many answers have been given. For simplicity, I shall focus on the one usually attributed to DeWitt [46], called the many-worlds view. Although that answer was frequently criticized as extravagant, it has the merit of being perhaps the clearest one.

In the many-worlds view, whenever there is a quantum measurement, the world in which the measurement is initiated splits into a large number of worlds.[7] There is at least one such world corresponding to each term in the right-hand side of (4). In world j, for instance, the quantum system ends up in state $\phi_j(\mathbf{x})$ and the apparatus in state $\alpha_j(\xi)$. That world henceforth continues to evolve according to the Schrödinger equation, in a way completely independent of the other ones.

This is Everett's solution to the measurement problem. There is no need for collapse because different readings occur in different worlds. Everett also shows that if the same quantum observable is repeatedly measured, every observer in every world will record that the results are repeated identically, just as quantum mechanics with collapse predicts.

Long-distance correlations are also explained quite straightforwardly in the many-worlds view. Suppose that two particles

[7] Some believe that splitting occurs whenever there is a quantum interaction, not necessarily one involving a macroscopic object.

have been prepared in the singlet state (5), and that Alice and Bob each have spin-measuring apparatus in initial states $|\alpha_0\rangle$ and $|\beta_0\rangle$, respectively. The compound system's initial state is thus given by

$$\frac{1}{\sqrt{2}}\{|+;\mathbf{n}\rangle|-;\mathbf{n}\rangle - |-;\mathbf{n}\rangle|+;\mathbf{n}\rangle\}\,|\alpha_0\rangle|\beta_0\rangle. \qquad (33)$$

After each particle has interacted with its measurement apparatus, the final state of the compound system is, in obvious notation, given by

$$\begin{aligned}\frac{1}{\sqrt{2}}\{&|+;\mathbf{n}\rangle|-;\mathbf{n}\rangle|\alpha_+\rangle|\beta_-\rangle \\ &- |-;\mathbf{n}\rangle|+;\mathbf{n}\rangle\}|\alpha_-\rangle|\beta_+\rangle. \qquad (34)\end{aligned}$$

The splitting into many worlds yields worlds where Alice's pointer shows $+$ and Bob's pointer shows $-$, and worlds where Alice's pointer shows $-$ and Bob's pointer shows $+$. There are no worlds where Alice's and Bob's pointers both show $+$, nor are there worlds where they both show $-$. Hence in all worlds, correlations predicted by standard quantum mechanics are perfectly satisfied.

VIII. Cramer's transactional interpretation

Cramer's transactional interpretation [17], [47] postulates that quantum processes (e.g., the emission of an alpha particle, followed by its absorption by one of several detectors) involve the exchange of offer waves (solutions of the Schrödinger equation) and confirmation waves (complex conjugates of the former). The confirmation waves propagate backward in time. Cramer's approach is inspired by the Wheeler-Feynman electromagnetic theory [35], [48], in which advanced electromagnetic waves are as important as retarded waves. The wave function and its complex conjugate are thus real fields, very much like the classical electric and magnetic fields.

Suppose that D, at point $\mathbf{x}$, is one of a number of detectors that can absorb the particle. The offer wave, emitted at t_0 from the alpha particle source, will arrive at D with an amplitude proportional to $\psi(\mathbf{x}, t)$, the Schrödinger wave function. The confirmation wave produced by D is stimulated by the offer wave, and Cramer argues that it arrives back at the source with an amplitude proportional to $\psi(\mathbf{x}, t)\psi^*(\mathbf{x}, t) = |\psi(\mathbf{x}, t)|^2$. Similar offer and confirmation waves are exchanged between the source and all potential detectors, and all confirmation waves reach the source exactly at t_0, the time of emission. Eventually, what Cramer calls a *transaction* is established between the source and one of the detectors, with a probability proportional to the amplitude of the associated confirmation wave at the source. The quantum process is then completed.

The transaction is, in Cramer's approach, what corresponds to the collapse of the wave function in standard quantum mechanics. Like collapse, the transaction picks just one of the pointer positions (which corresponds, in our example, to the detector that has fired). But unlike collapse, the transaction does not occur at a specific time. It occurs on the whole space-time region that links the source and the detector, in what Cramer calls *pseudotime*.

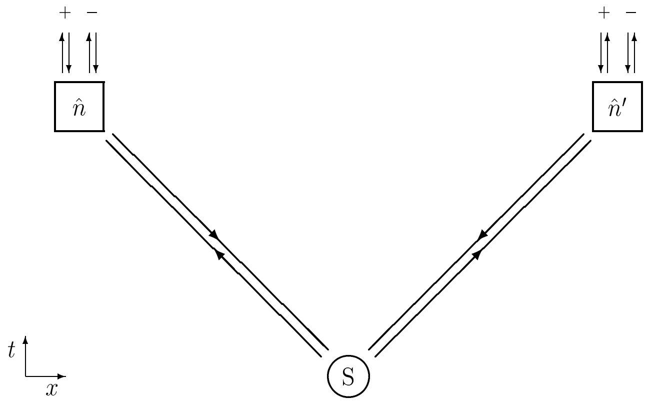


Fig. 3. Offer waves (upward arrows) and confirmation waves (downward arrows) in the EPR setup.

The transactional interpretation provides a rather vivid representation of the mechanism of long-distance correlations. Fig. 3 is a space-time representation of an EPR setup, viewed in the transactional interpretation [42]. Arrows pointing in the positive time direction label offer waves, and those pointing in the negative direction label confirmation waves. Two particles are emitted by the source, and in Cramer's sense both Alice's and Bob's particles can be absorbed by two detectors. They correspond to the two beams in which each particle can emerge upon leaving its spin-measuring device.

Let us focus on what happens on the left-hand side. An offer wave is emitted by the source, and in going through the spin-measuring device it splits into two parts. One part goes into the detector labelled $+$, and the other goes into detector $-$. Each detector sends back a confirmation wave, propagating backward in time through the apparatus and reaching the source at the time of emission. A transaction is eventually established, resulting in one of the detectors registering the particle. A similar process occurs on the right-hand side, with one of the two detectors on that side eventually registering the associated particle.

If offer and confirmation waves represent a special kind of causal influences, one can see that these influences can be transmitted between the spacelike-separated detectors on different sides along paths that are entirely timelike or lightlike. The EPR correlations are thus explained without introducing any kind of superluminal motion, which is one more way to meet Feynman's challenge.

IX. Discussion

Bohmian mechanics, the many-worlds view, and the transactional interpretation are three possible answers to the question of how can the world be for quantum mechanics to be true. Bohmian mechanics tells us that microscopic particles follow deterministic trajectories influenced by the quantum potential. The many-worlds view asserts that all results of a quantum measurement simultaneously exist, but in different worlds that cannot communicate with each others. The transactional interpretation tells us that backward-in-time connections are effected through the complex conjugate Schrödinger field, and that transactions are established between emitters and specific detectors.

Of course these interpretations of quantum mechanics, just like others that I have not considered explicitly, also have problems, since none of them has gained universal acceptance. Bohmian mechanics, for instance, is not easy to reconcile with the theory of special relativity. The many-worlds view is often deemed extravagant, while more benign implementation of Everett's approach may not be so well-defined. And the notion of transaction needs to be spelled out more precisely.

Apart from specific criticisms, the whole program of interpreting quantum mechanics has been questioned by adherents of the epistemic view. Why bring forward interpretations that add no empirical content to the theory? If, for instance, Bohmian mechanics exactly reproduces the statistical results of quantum mechanics, aren't the trajectories superfluous, and shouldn't they be discarded? The analogy has been made between such trajectories and the concept of the ether prevalent at the turn of the twentieth century [49], [50]. H. A. Lorentz and his contemporaries viewed electromagnetic phenomena as taking place in a hypothetical medium called the ether. From this, Lorentz developed a description of electromagnetism in moving reference frames, and he found that the motion is undetectable. Following Einstein's formulation of the electrodynamics of moving bodies, the ether was recognized as playing no role, and was henceforth discarded. So should it be, according to most proponents of the epistemic view of quantum states, with interpretations of quantum mechanics that posit observer-independent elements of reality like Bohmian trajectories. They predict no empirical differences with the Hilbert space formalism, and should therefore be discarded.

It is true that, just like the ether in special relativity, constructs like Bohmian trajectories don't lead to specific empirical consequences. I have argued, however, that although they could be dispensed with in the hypothetical world of Sec. IV, they cannot in the real world unless, just like the ether was eventually replaced by the free-standing electromagnetic field, they are replaced by something that can account for the structure of macroscopic objects.

In all physical theories other than quantum mechanics, there are straightforward and credible answers to the question raised above, of "How can the world be for the theory to be true?" In quantum mechanics there are a number of answers, like the ones we have reviewed in this paper. None is straightforward, and none gains universal credibility. Should we then adopt the attitude of the epistemic or related views, which decide not to answer the question? I believe that, from a foundational point of view, this is not tenable. For how can we believe in a theory, if we are not prepared to believe in any of the ways it can be true, or worse, if we do not know any way that it can be true?

The epistemic view of quantum mechanics is an attempt to solve or attenuate the foundational problems of the theory. It would succeed if quantum mechanics were used only to explain nonclassical correlations between macroscopic objects. But it is also used to explain the microscopic structure of such objects. Interpreting the theory means finding ways that it can be intelligible. I believe that each clear and well-defined way to do so adds to the understanding of the theory. In many instances, however, much work remains to be done to achieve that clarity and precision.

Acknowledgment

This work was supported by the Natural Sciences and Engineering Research Council of Canada.

References

[1] L. Marchildon, "Does quantum mechanics need interpretation?" *Proceedings of the Third International Conference on Quantum, Nano and Micro Technologies*, D. Avis, C. Kollmitzer, and V. Privman, Eds. Los Alamitos, CA: IEEE, 2009, pp. 11–16.

[2] T. M. Nieuwenhuizen, B. Mehmani, V. Spicka, M. J. Aghdami, and A. Y. Khrennikov, Eds. *Proceedings of the Beyond the Quantum Workshop*. Singapore: World Scientific, 2007.

[3] L. Accardi, G. Adenier, C. Fuchs, G. Jaeger, A. Y. Khrennikov, J. A. Larsson, and S. Stenholm, Eds. *Foundations of Probability and Physics - 5*. AIP Conference Proceedings 1101, Berlin: Springer, 2009.

[4] G. C. Ghirardi, "The interpretation of quantum mechanics: where do we stand?" *Journal of Physics: Conference Series*, 174: 012013, 1–16 (2009).

[5] M. Genovese, "Research on hidden variable theories: a review of recent progresses," *Physics Reports*, 413: 319–396 (2005).

[6] A. Einstein, B. Podolsky, and N. Rosen, "Can quantum-mechanical description of physical reality be considered complete?" *Physical Review*, 47: 777–780, May 1935.

[7] C. H. Bennett and G. Brassard, "Quantum cryptography: public key distribution and coin tossing," *Proceedings of the IEEE International Conference on Computers, Systems and Signal Processing*. New York: IEEE, 1984, pp. 175–179.

[8] P. W. Shor, "Algorithms for quantum computation: discrete logarithms and factoring," *Proceedings of the 35th Annual Symposium on Foundations of Computer Science*, S. Goldwasser, Ed. Los Alamitos, CA: IEEE, 1994, pp. 124–134.

[9] G. 't Hooft, "Quantum gravity as a dissipative deterministic system," *Classical and Quantum Gravity*, 16: 3263–3279, October 1999.

[10] A. J. Leggett, "Testing the limits of quantum mechanics: motivation, state of play, prospects," *Journal of Physics: Condensed Matter*, 14: R415–R451, April 2002.

[11] C. Rovelli, "Relational quantum mechanics," *International Journal of Theoretical Physics*, 35: 1637–1678, August 1996.

[12] C. A. Fuchs and A. Peres, "Quantum theory needs no 'interpretation'," *Physics Today*, 53: 70–71, March 2000.

[13] C. A. Fuchs, "Quantum mechanics as quantum information (and only a little more)," in *Quantum Theory: Reconsideration of Foundations*, A. Khrennikov, Ed. Växjö: Växjö U. Press, 2002, pp. 463–543. Also available as quant-ph/0205039.

[14] P. E. Vermaas, *A Philosopher's Understanding of Quantum Mechanics. Possibilities and Impossibilities of a Modal Interpretation*. Cambridge: Cambridge U. Press, 1999.

[15] D. Bohm, "A suggested interpretation of the quantum theory in terms of 'hidden' variables (I and II)," *Physical Review*, 85: 166–193, January 1952.

[16] H. Everett III, " 'Relative state' formulation of quantum mechanics," *Reviews of Modern Physics*, 29: 454–462, July 1957.

[17] J. G. Cramer, "The transactional interpretation of quantum mechanics," *Reviews of Modern Physics*, 58: 647–687, July 1986.

[18] L. Marchildon, "Bohmian trajectories and the ether: where does the analogy fail?" *Studies in History and Philosophy of Modern Physics*, 37: 263–274, June 2006.

[19] L. Marchildon, "The epistemic view of quantum states and the ether," *Canadian Journal of Physics*, 84: 523–529, January 2006.

[20] L. Marchildon, "Why should we interpret quantum mechanics?" *Foundations of Physics*, 34: 1453–1466, October 2004.

[21] M. Jammer, *The Conceptual Development of Quantum Mechanics*, 2nd ed. Tomash/American Institute of Physics, 1989.

[22] L. E. Ballentine, "The statistical interpretation of quantum mechanics," *Reviews of Modern Physics*, 42: 358-381, October 1970.

[23] L. Marchildon, *Quantum Mechanics. From Basic Principles to Numerical Methods and Applications*. Berlin: Springer, 2002.

[24] J. von Neumann, *Mathematical Foundations of Quantum Mechanics*. Princeton: Princeton U. Press, 1955.

[25] G. C. Ghirardi, P. Pearle, and A. Rimini, "Markov processes in Hilbert space and continuous spontaneous localization of systems of identical particles," *Physical Review A*, 42: 78–89, July 1990.

[26] V. Privman and D. Mozyrsky, "Decoherence and measurement in open quantum systems," *SPIE Proceedings*, E. Donkor and A. R. Pirich, Eds. 4047: 36–47 (2000).

[27] M. Xiao, I. Martin, E. Yablonovitch, and H. W. Jiang, "Electrical detection of the spin resonance of a single electron in a silicon field-effect transistor," *Nature*, 430: 435–439, 22 July 2004.

[28] M. Zwolak, H. T. Quan, and W. H. Zurek, "Quantum Darwinism in a hazy environment," arXiv:0904.0418v1.

[29] I. Bloch, "Some relativistic oddities in the quantum theory of observation," *Physical Review*, 156: 1377–1384, April 1967.

[30] W. Heisenberg, *Physics and Philosophy. The Revolution in Modern Science*. New York: Harper, 1958.

[31] R. Peierls, "In defence of 'measurement'," *Physics World*, 4: 19–20, January 1991.

[32] O. Ulfbeck and A. Bohr, "Genuine fortuitousness. Where did that click come from?" *Foundations of Physics*, 31: 757–774, May 2001.

[33] A. Bohr, B. R. Mottelson, and O. Ulfbeck, "The principle underlying quantum mechanics," *Foundations of Physics*, 34: 405–417, March 2004.

[34] N. D. Mermin, "What is quantum mechanics trying to tell us?" *American Journal of Physics*, 66: 753–767, September 1998.

[35] J. A. Wheeler and R. P. Feynman, "Classical electrodynamics in terms of direct interparticle action," *Reviews of Modern Physics*, 21: 425–433, July 1949.

[36] R. P. Feynman, *The Character of Physical Law*. Cambridge, MA: MIT Press, 1967.

[37] R. N. Giere, *Explaining Science. A Cognitive Approach*. Chicago: U. of Chicago Press, 1988.

[38] F. Suppe, *The Semantic Conception of Theories and Scientific Realism*. Urbana: U. of Illinois Press, 1989.

[39] D. Bohm and B. J. Hiley, *The Undivided Universe*. London: Routledge, 1993.

[40] L. de Broglie, "La mécanique ondulatoire et la structure atomique de la matière et du rayonnement," *Journal de physique*, série VI, 8: 225–241, May 1927.

[41] C. Philippidis, C. Dewdney, and B. J. Hiley, "Quantum interference and the quantum potential," *Il Nuovo Cimento*, 52B: 15–28, July 1979.

[42] L. Marchildon, "Understanding long-distance quantum correlations," in [2], pp. 155–62.

[43] C. Dewdney, P. R. Holland, and A. Kyprianidis, "What happens in a spin measurement?" *Physics Letters A*, 119: 259–267, December 1986.

[44] C. Dewdney, P. R. Holland, and A. Kyprianidis, "A causal account of non-local Einstein-Podolsky-Rosen spin correlations," *Journal of Physics A: Mathematical and General*, 20: 4717–4732, October 1987.

[45] A. Kent, "One world versus many: the inadequacy of Everettian accounts of evolution, probability, and scientific confirmation," arXiv:-0905.0624v1.

[46] B. S. DeWitt, "Quantum mechanics and reality," *Physics Today*, 23: 30–35, September 1970.

[47] J. G. Cramer, "Generalized absorber theory and the Einstein-Podolsky-Rosen paradox," *Physical Review D*, 22: 362–376, July 1980.

[48] J. A. Wheeler and R. P. Feynman, "Interaction with the absorber as the mechanism of radiation," *Reviews of Modern Physics*, 17: 157–181, April-July 1945.

[49] J. Bub, "Why the quantum?" *Studies in History and Philosophy of Modern Physics*, 35: 241–266, June 2004.

[50] J. Bub, "Quantum mechanics is about quantum information," *Foundations of Physics*, 35: 541–560, April 2005.

Unified Vectorization of Numerical and Textual Data using Self-Organizing Map

Farid Bourennani, Ken Q. Pu, Ying Zhu
University of Ontario Institute of Technology, Canada
{farid.bourennani, ken.pu, ying.zhu}@uoit.ca

Abstract—**Data integration is the problem of combining data residing in different sources, and providing the user with a unified view of these data. One of the critical issues of data integration is the detection of similar entities based on the content. This complexity is due to three factors: the data type of the databases are heterogeneous, the schema of databases are unfamiliar and heterogenous as well, and the quantity of records is voluminous and time consuming to analyze. Firstly, in order to accommodate the textual and numerical heterogeneous data types we propose a new weighting measure for the numerical data type called Bin Frequency - Inverse Document Bin Frequency (BF-IDBF). Our proposed BF-IDBF measure is more efficient than histograms, when combined with Term Frequency - Inverse Document Frequency (TF-IDF) measure for Heterogeneous Data Mining (HDM) by Unified Vectorization (UV). The UV permits to combine the algebraic models representing heterogeneous data documents, e.g. textual and numerical, which make the simultaneous HDM process simpler and faster than the traditional attempts to process data sequentially by their respective data type. Secondly, in order to handle the unfamiliar data structure, we use the unsupervised algorithm, Self-Organizing Map (SOM). Finally to help the user to explore and browse the semantically similar entities among the copious amounts of data, we use a SOM-based visualization tool to map the database entities based on their semantical content.**

Index Terms—**Pre-Processing, Data Integration, Heterogeneous Data Mining (HDM), Unified Vectorization (UV), Self Organizing Map (SOM).**

I. Introduction

Many industrial sectors such as finance, medicine, data integration, and others are very interested in heterogeneous data classification. This interest is proportional to the heterogeneity of the data types. In other words, the more the data types are heterogeneous, the higher is the motivation for heterogeneous data mining in order extract convergent results from these dissimilar data types.

In the data integration context, the purpose for joining multiple databases is to significantly increase the data richness. However, due to the large volume of data (terabytes), an automated support is needed in order to find semantic matches between database entities located in two or more data sources. Here, database entities are in fact tables' columns in a relational database model. The schema matching operation is complex due to the heterogeneities found in the different databases, such as the heterogeneity in the data types or data models. In this paper, we extend our previous works [1], [2] to overcome these heterogeneities found in different unfamiliar data repositories, and integrate the data in an efficient manner.

One of the qualities of a properly integrated data is a tight coupling. A tightly coupled distributed DB system provides location, replication and distribution transparencies to the user. Consequently, in order to achieve a tight coupling, the complete availability of databases documentation is necessary for the developer to understand the heterogeneous databases. Very often the information on the data schemas is not available because it is located in different locations or it has never been completed. In that case, exploiting semantic content to determine automatically similar database entities is the only way to achieve a tight coupling [3]. Firstly, the operation of finding database entities having similar content is done manually by developers. Manually specifying schema matches is a tedious, time-consuming, error-prone, and therefore expensive process [4]. Secondly, even after determining schema matches, the problem with the current integration tools is that they do not scale well to large schemas, and yet that is exactly what business need [5]. To solve this problems, in this paper we propose to classify automatically, by using SOM [6], the database entities based on the content in order to realize a tight coupling. The visualization properties of SOM make our tool very scalable to large schemas.

In brief, in this paper we use the SOM based visualization tool for data integration purposes with a focus on the pre-processing phase. The proposed pre-processing techniques permit to process simultaneously heterogeneous textual and numerical data types. The resulting SOM map provides to the user a unified view of the data entities despite the heterogeneity of the data types. Furthermore, the map topology reflects the content similarities between database entities, which make data integration operation scalable and the data tightly coupled.

Contribution of the paper

- We show that by combining TF-IDF and BF-IDBF measures, for heterogeneous data mining by unified vectorization, it is possible extract *more convergent mining results*, from *heterogeneous* textual and numerical data types, than the combination of TF-IDF and histograms measures.
- We demonstrate how the unified vectorization of numerical and textual data, for SOM-based processing, can facilitate the schema matching operations by overcoming the different heterogeneities found in the distributed databases.
- We illustrate that the heterogeneous data mining by combining the TF-IDF and BF-IDBF measures is more precise and faster than the usage of post-processing algorithms on

larger data sets, which is demonstrated by new experiments.

- We demonstrate that the SOM-based visualization tool is more appropriate to large database schemas than ontology based tools, for explorative purposes, because of its scalable properties.

Outline of the paper

The structure of the rest of the paper is as follows. Section II, III, IV, and V will discuss the Review, the Pre-Processing, the Processing, and the Post-Processing phases respectively. The section VI is devoted to experiments while section VII concludes the present paper.

II. Review

The definition of data integration: it is the process of combining data residing in different data sources and providing users with a unified view of these data [7]. Many automatic schema mapping techniques have been proposed by researchers [4][8][9][10][11] to facilitate this task. These automated tools can be divided into two main categories: Ontology based vs. semantic based data integration. The ontology based tools emphasize on database structural model, while semantic based integration tools accent more on the semantical content of database entities. The big majority of the existing data integration tools are are ontology based, however some research groups, such as Microsoft, are exploring the use of semantical content for this purpose as well [5]. Probably the ontology based tools are more used because the purpose of data integration is to build a new data warehouse, which implies a new data model. It is logical to use tools based on database structural models in order to build new database models. However, the problem is that before matching these database entities, the developer needs to find manually the *similar* ones. That is exactly where the ontology tools are lacking, and where the semantic based tools have their strength by automatically extracting similar database entities using information retrieval techniques for example.

The ontology based tools use mainly database structural model. And, the majority of the current tools are ontology based, which means that the schema matching is done manually[5]. As mentioned previously, manually specifying schema matches is a tedious, time-consuming, error-prone, and therefore expensive process [4]. In addition, it is extremely time consuming to detect semantical relations between entities using ontologies because of the huge amount of data, the semantic conflict, and the unavailable documentation on the database schemas [5] [3].

As explained earlier, the semantic based integration tools are based on semantical content of database entities. Salton [12] and Van Rijsbergen [13] were the first ones to introduce textual information retrieval techniques for classifying textual databases based on the semantics. However, for some reason these kind of tools are not used for data integration. Probably, these type of tools are not used because they are not that practical without visualization features added to them. Another possible reason for not using these methods is that they do not provide a unified view of the heterogeneous data types. Some research groups, such as *Microsoft*, expressed their interest in these kind of tools, but it remains that they are not commercialized yet. Indeed, the use of these content based techniques is necessary in order to build automated tools for data integration purposes, and realize a tight coupling. Because, it is firstly necessary to detect similar database entities based on the semantic content by using tools such as the one proposed in this paper. Then, once these similarities are detected, the process of combining data residing at different sources using the traditional ontology based tools is more appropriate because it serves better this purpose.

However, in order to find these semantically related database entities located in different repositories, a couple of issues need to be addressed. Sheth [14] groups these concerns into "heterogeneities". More precisely, he divides them into four groups: syntactic heterogeneity, schematic or structural heterogeneity, representational heterogeneity, and semantic heterogeneity. Every one of these heterogeneities or issues is described bellow. In addition, we explain which solution do we propose to each one of these heterogeneities.

- *Syntactic heterogeneity* comprises all aspects that are related to the specific technical choices for the representation of interfaces and data. For example, two databases could use two different relational database management systems (RDBMS) such as Oracle vs. DB2. Consequently, in order to query or extract database entities from these syntactically dissimilar repositories, different layers or interfaces need to be used for every database. Rather than developing complex layers, queries, and interfaces, in this project we propose to uniformly transform all the relational database columns, from these heterogeneous and unfamiliar repositories, into uniform files, e.g. text files. In other words, every text file is a database column that can be further, uniformly, processed for data integration purposes.
- *Schematic or structural heterogeneity* means differences in the types and structures of the elements. Many research groups are interested in heterogeneous data mining, and this interest is proportional to the heterogeneity of the data. The more the data is heteregneous, the higher is the interest to HDM. For example, in the business world several projects [15][16][17][18][19][20] worked on heterogeneous textual and numerical data mining because of the availability of the data. These projects focused on the mining combination of two data types in order to enhance the quality of the extracted information and the classification results. As an illustration, for the textual data, it could be business reports, and for the numerical data, it can be stock market prices. Putting together the mining results will provide much more valuable information. However in all these research projects, the resulting clusters from the qualitative and quantitative analysis did not coincide, rather they diverged and the obtained results were of lower quality. Most probably, the main reason of this divergence is caused by the mining results combination. I.e., each data type was processed separately, then the complex combination of these outputs led to poorer quality of

clusters. In this research it is proposed to extract a more convergent classification results by mining numerical and textual data types *simultaneously* by Unified Vectorization (UV). By using UV, it is possible to combine the two data types in order to simultaneously process them. Furthermore, it permits to extract more convergent mining results, from heterogeneous textual and numerical data types, and avoid the complex combination of the classification results. In addition, it allows to have a unified view all the data regardless of their type for data integration purposes. Finally, it opens the doors for mining other data types in the future.

- *Model or representational heterogeneity* implies differences in the underlying models (database, ontologies) or their representations (relational, object-oriented, RDF, OWL). These kind of heterogeneities are usually handled by ontology based tools. In this paper, we prefer to focus mainly on the content of the database rather than on the models because it is too complicated to process automatically. In addition, even if automated tools based on the models are developed, nevertheless the content of the database entities needs be examined, manually or automatically, in order validate the similarities between database entities. That is why, as explained previously, semantic based tools are necessary for data integration purposes, and tight coupling. To solve this issue, we propose, similarly to the syntactic heterogeneity problem, to transform all the database entities from different data sources into unstructured files of the same type, hence, text files for example. Then, these files are processed uniformly by an unsupervised algorithm, in this case Self Organizing Map (SOM), for detecting semantical content similarity between the databases entities. The resulted map will facilitate the user to explore visually the relations between database columns (entities) despite difference of data models or representations. Even if the data schemas are ignored in the processing, a certain portion of the original data models is shown, on the SOM's map, in the data entities labeling. The database entities labels, on the SOM's map, have this format: column@table, which represents the column name, and the table name to which this column belongs to. Another reason for doing the classification of the database entities purely based on the columns' content, and making an abstraction on the databases respective models, is that they are difficult to scale when schemas are large, and yet that is exactly what business need[5].
- *Semantic heterogeneity* connotes where the same real world entity is represented using different terms or vice-versa. For example, if two database are merged, some definitions and concepts in their respective schemas like "earnings" may have different meanings. in one database, "earnings" may mean profits in dollars, while in the other database it might be the number of sales. Another example could be the concept of "clients", which is called "clients" in one database and "costumers" in the other one. The information in these database entities is similar, but the terminology used is different. Ontology of cross-lexical references, such as Wordnet [21], could be used to address this problem by finding synonyms. However the processing would be heavier, and the content of traditional databases is too basic to use these kind of techniques. In this paper, we propose to process database entities' content by using a more advanced text vectorization technique, called N-gram[25]. Actually, the N-Grams offers the advantage of detecting similarities between two terms despite typo mistakes or two different words having similar roots. For more details and examples, refer to next section.

III. Pre-processing

In order to implement any classification method, it is necessary to transform the input documents into an algebraic model so they can be processed. In this paper, the documents are in fact database entities, more specifically columns in a relational database model.

The standard practice in information retrieval is the usage of the vector space model (VSM) to represent text documents [22]. Documents are symbolized in t-dimensional Euclidean space where each dimension corresponds to a term of the vocabulary [12]. Despite its simplicity and efficiency, the VSM has the disadvantage of focusing only on the textual data type. First, it's hard to obtain accurate information of semantic relatedness automatically from textual information only [23]. And, more complex is the extraction of convergent results or coincident meaning from heterogenous data types.[16]

Therefore, as shown on Fig. 1, our approach proposes to pre-process the heterogeneous data types, such as numerical and textual data, separately. Then combine them for simultaneous data mining by *unified vectorization* for better and more convergent content based clustering results.

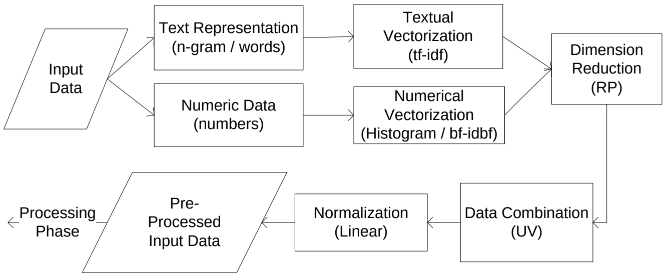


Fig. 1. Preprocessing Phase

A. Pre-Processing of the Textual Data

Let us start with textual data first and use the mothodology as a reference for numerical data, because numbers are texts, but the opposite is false. Initially, all the textual portions of a column d_j are transformed into a vector x_i. Thus, the combination of all the x_i will form the textual portion of the VSM. The composition of x_i is done as follow:

$$x_i = (w_{1j}, w_{2j}, ...w_{|T|j}),$$

where $|T|$ is the number of terms in whole set of terms T, and w_{kj} represents the vectorization or the weight of the term t_k in the document(column) d_j.

Several text representations are mentionned in the litterature, the most important ones are: Bag of Words [24], N-gram[25], Stemming and Lemmatization [24]. In this research paper, "bag of words" and N-gram text representations are used. Firstly, the most common text representation within VSM framework is "bag of words" [24] [22]. Every word is a term in the VSM model. Secondly, another efficient text representation is N-grams, which is a substring of N consecutive characters of word. For a given document, all the N-grams are in fact all the terms that compose the VSM matrix. N-gram has been chosen because it offers several advantages; it's an easy and fast way to solve syntax related issues such as misspelling. Moreover, it finds common patterns between words with the same roots, but with different morphological forms (e.g., finance and financial), without treating them as equal, which happens with word stemming [26].
Example:

Suppose that a document contains these two words: *Finance* and *Financial*.

4 - Gram of "finance" $\Longrightarrow$ **fina, inan, nanc**, ance

4 - Gram of "financial" $\Longrightarrow$ **fina, inan, nanc**, anci, cial
Even if the words are different, they have three common tokens, which reflects their semantical similarity.

A.1 Vectorization: TF-IDF measure

Most approaches[24][27] are centered on a vectorial representation of texts using Term Frequency - Inverse Term Frequency(TF-IDF) measure, defined as:

$$\text{TF-IDF}(t_k, d_j) = \frac{Freq(t_k, d_j)}{\sum_k Freq(t_k, d_j)} \times Log\frac{N_{\text{doc}}}{N_{\text{doc}}(t_k)}$$

where $Freq(t_k, d_j)$ denotes the number of times the term t_k occurs in the document (column) d_j, $\sum_k Freq(t_k, d_j)$ is the total number of all the term occurrences in the same document d_j, and N_{doc} is the total number of documents in the corpus, while $N_{\text{doc}}(t_k)$ is the number of documents in the corpus with the term t_k.
Example:
- Suppose the corpus D composed of group of 5 documents D={d1, d2 , d3 , d4 , d5}.
- The content of d1 is: "Hello World"
- Suppose all documents have the word: hello
  TF - IDF (hello, d1) = (1 / 2) * log (5 / 5) = 0
- If 4 documents only have the word: hello
  TF-IDF (hello, d1) = (1 / 2) * log (5 / 4) = 0.048
- Suppose d1, d2 , d3 have the word: world
  TF-IDF (world, d1) = (1 2) * log (5 / 3) = 0.255

The three last examples illustrate the smoothing function of the log in the TF-IDF formula.

A.2 Dimensionality reduction

In the text applications context, the high dimensionality is due to the large vocabulary of the corpus, which includes in addition to regular words, names, abbreviations, and others. This high number to terms (tokens) leads to burdensome computations and even restricts the choice of data processing methods. A statistical optimum of dimensionality reduction is to project the data onto a lower-dimensional orthogonal subspace that captures as much of the variation of the data as possible. The most widely used way to do this is Principal Component Analysis (PCA), however it is computationally expensive and is not feasible on large, high-dimensional data[28]. Therefore, another powerful technique that solves these problems is the Random Projection (RP) which is simple, offers clear computational advantages, and preserves similarity[28]. Given a matrix X, the dimensionality of the data can be reduced by projecting it through the origin onto a lower-dimensional subspace, formed by a set of random vectors R:

$$A_{[k \times n]} = R_{[k \times m]} \bullet X_{[m \times n]}$$

Where, A is the reduced matrix, and the k in the subscripts is the desired, reduced dimensionality.

Random Projection method was successfully tested with SOM on text and image data types, and it appears to be a good alternative to traditional methods of dimensionality reduction particularly when the dimension gets large [28]. More specifically, with textual data RP seems to perform better when the reduced dimensionality is superior to 600. The difference between the average error of RP and SVD (PCA performed directly on the data matrix) is less than 0.025 with 95% confidence interval [29].

B. Treatment of Numerical Data:

Because of the different nature of data, the numerical data is pre-processed differently from the textual one. As an illustration let us have two numbers 1988 and 1991, representing years of birth or financial values, present in two columns (documents). Their proximity will not be detected by using traditional textual representations such as bag of words or n-gram because they do not possess enough textual similarities. That is why it is essential to pre-process the numeric input data differently so that their vectorized values reflects their semantic similarity.

Several techniques are mentioned in the literature to specify concept hierarchies for numerical attributes such as binning, histogram analysis, entropy-based discretization, Z2-merging, cluster analysis and discretization by intuitive partitioning[30].

In this research, Histogram analysis [30] is used to ease the SOM neural network's learning process and improve the quality of the map. More precisely, Equal-Frequency (Equal-Depth) Histogram [30] is used because of its good scaling properties and simplicity to implement. The values are partitioned so that, ideally, each partition, called bucket or bin, contains identical number of tuples. Another good reason for using histogram is that it reduces the dimensionality of the numerical portion of the VSM by s times, where s is the size of the bin. However, sometimes the size of the bins can be bigger than s in order to avoid cutting a cluster of the same value in the middle because of trying to respect the bin size. In this paper, the histogram bin (bucket) size was fixed to 10. All the numerical data n_i of the document d_j are transformed into a vector n_i:

$$n_i = (v_{1j}, v_{2j}, ...v_{|N|j}),$$

where, $|N|$ is the total number of histogram bins, and v_{lj} represents the number of observations that fall into various disjoint bin b_l.

The combination of all the vectors n_i will form the VSM of the numerical data type as shown on Tab. 1.

Example:
- Suppose that these numbers {1, 2.5, 3, 3, 4, 5, 7, 9} are in the corpus D, and the bin size is equal to 3.
- Bin1 = {1, 2.5, 3, 3}
- Bin2 = {4,5,7}
- Bin3 ={9}

- Assume that a document *d1*, where *d1* $\in$ *D*, has the numbers {1, 3} in his content.
- Therefore, Hist(Bin1, d1) = 2

C. *Combination of the textual and the numerical data types by Unified Vectorization*

Now that the textual and numerical data have been vectorized and the dimensionality reduced, it is proposed to combine the numerical and the textual data by Unified Vectorization, as shown on Tab. 1, for simultaneous data processing. This combination of heterogeneous data types permits to meet the challenge of extracting convergent classification results out of this *heterogeneous* data types. However, the unified VSM should be normalized in order to avoid any unjustified influence of one the two data types during the SOM training phase.

Tab. 1 Unified Vectorization of Textual and Numerical Data Types

Docs	Terms				Bins			
	t_1	t_2	$\dots$	t_n	b_1	b_2	$\dots$	b_l
d_1	w_{11}	w_{12}	$\dots$	w_{1n}	v_{11}	v_{12}	$\dots$	v_{1k}
d_2	w_{21}	w_{22}	$\dots$	w_{2n}	v_{21}	v_{22}	$\dots$	v_{2k}
$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$
d_m	w_{m1}	w_{m2}	$\dots$	w_{mn}	v_{m1}	v_{m2}	$\dots$	v_{mk}

D. *Heterogenous data processing optimization*

The HDM by UV, using SOM, offers good results despite the difficulty of extracting convergent results from the heterogeneous textual and numerical data types[1]. However, the problem is that the results were not good enough in our first experiments. For example, it is well known that using N-gram as tokenizer for textual data gives better results than "bag of words" [25]. But surprisingly when unified vectorization was applied using N-grams and histograms, the results were poorer than the combination of "bag of words" and histograms. This unsatisfactory performance of the N-gram tokenizer can be explained by the dissimilarity of the vectorization techniques. In other words, the TF-IDF measure for textual data and histogram measure for the numerical data don't have an equivalent representation of tokens. That is why unexpectedly "bag of words" performs better than n-gram. Indeed, TF-IDF measures the importance of a token in the document as well as its general importance in the corpus. On the contrary, histogram measures only the importance of a token in the document, while its importance in the corpus is neglected.

In order to solve this insufficiency in the results, two alternatives are explored. The first one, is to find a better way to vectorize the numerical data, so that it represents a similar type of information as the text TF-IDF measure. In this direction, BF-IDBF measure of numerical data is introduced for better HDM. Another solution, is to hide the dissimilarity between two measures(TF-IDF vs. Histogram) by post-processing the resulted VSM matrix. Consequently in this sense, we use a post-processing algorithm called Common Item-set Based Classifier (CIBC) [1].

E. *BF-IDBF weighting*

In spite of good results with HDM by UV, the usage of histograms as representation for numerical data type was was not as good as expected when it was combined with the TF-IDF measure for textual data. Probably, the reason is that in the opposite of the TF-IDF measure, the histogram measure does not give a sense of rarity or importance of a number in the corpus. In other words the two representation are not equivalent because they don't reflect exactly the same type of information. More specifically, it is the IDF component that is not represented in the histogram measure.

To solve this problem we propose the usage of Bin Frequency - Inverse Document Bin Frequency (BF-IDBF) weight as an alternative for representing the numerical data type. Actually, BF-IDBF model has two advantages. First, it uses the properties of an histogram which are more appropriate for numerical data type due to the different nature of the data. Secondly, it offers a data representation that is equivalent to TF-IDF measure in consequence of which the Machine Learning (ML) algorithms perform better when the data is processed by Unified Vectorization. In other words, BF-IDBF is an equivalent measure to the TF-IDF and it is based on the histogram at the same time.

First the histogram is computed, then the BF-IDBF measure is be calculated in two step, the BF, then the IDBF. The BF serves to estimate the importance of bin, rather than the importance of a number, in a document. This way, it is possible to benefit from the precision and simplicity that offers Equal-Depth histograms. Likewise, the bin reduces the number of terms in the VSM matrix, which simplifies significantly the processing time and resources.

The BF is defined as follows:

$$\text{BF}(b_l, d_j) = \frac{Freq(b_l, d_j)}{\sum_k Freq(b_l, d_j)}$$

where, $Freq(b_l, d_j)$ denotes the number of times the bin b_l occurs in the document (column) d_j, $\sum_k Freq(b_l, d_j)$ is the total number of all the bin occurrences in the same document d_j.

It is important to mention that in this paper for simplicity reason the bin size b_l was fixed ideally to 10. This number can sometimes vary in order to keep in the same bucket the numbers that are equal. Probably, a more efficient method for determining bucket sizes would improve the classification results. More details about those methods can be found in [31].

Other variances of histogram could be used as well, but because of the good results obtained in the previous work [1], "equal depth" histogram is kept.

The next step is the calculation of the IDBF weight which mainly serves to reduce the weight of the bin when it is not important in the corpus. In other words, if a certain range of numbers are common to a high amount of documents, the weight is decreased for better document classification based on the numerical semantical content. The IDBF is defined through a similar formula to IDF calculation as follow:

$$\text{IDBF}(b_l, d_j) = Log \frac{N_{\text{doc}}}{N_{\text{doc}}(b_l)}$$

where N_{doc} is the total number of documents in the corpus, while $N_{\text{doc}}(b_l)$ is the number of documents in the corpus with the bin b_l.

Finally, the BF-IDBF is calculated by multiplying the two measures, therefore the global formula is:

$$\text{BF-IDBF}(b_l, d_j) = \frac{Freq(b_l, d_j)}{\sum_k Freq(b_l, d_j)} \times Log \frac{N_{\text{doc}}}{N_{\text{doc}}(b_l)}$$

F. Normalization

The data does not necessarily have to be pre-processed at all before creating SOM and using it. However, in most real tasks pre-processing is important; perhaps even the most important part of the whole process[32]. All the values of the unified VSM matrix were normalized similarly in a range of [0,1] through a linear operation.

IV. Processing

Unsupervised classification or "clustering" is one of the fundamental data mining techniques. Furthermore, Self Organizing Map (SOM) is an unsupervised learning neural network that produces a topologically clusters mapping on a plane (2D). The unsupervised classification property of SOM serves to classify completely unfamiliar database entities. In essence, despite the unknown databases schemas, the different database respective technologies, the different entities naming standards (client vs. customer), it would be possible to integrate these databases based on their semantic content by using SOM. Other unsupervised algorithms such as ART [35] could be used as well, but the SOM's trained map conveys additional information beyond the strict clustering of input documents. The SOM's map topology reflects the content similarity between documents. In addition, the SOM algorithm is computationaly simple and produces reasonable results when compared to ART2 [36].

A. Self Organizing Map

Self Organizing Map (SOM) of Kohonen is an unsupervised learning method which is based on the principle of competition according to an iterative process of updates[37]. It was used in numerous work in visualization of text corpus and applied in thousands of research projects[23]. SOM has two training modes that are mentioned in the literature: sequential and batch version. They differ basically in the method of updating weight vectors. The batch method of SOM was preferred over the sequential version because of two reasons. a) it produces a map much faster and b) it does not need a learning rate to converge. More details can be found in [6][23][33].

In brief, the batch version of SOM works as follows:

- *Phase 1:* Compare each input vector d_j with all the map nodes m_i, initially selected randomly, in order to find the best matching unit (BMU). Then, copy each d_j into a sub-list associated with that map unit.
- *Phase 2:* When the entire d_j have been distributed into their respective BMUs sub-lists in this way, consider the neighborhood set N_i, around the map unit m_i. In other words N_i represents all the map units within the radius of map unit m_i. In the union of all sub lists in N_i, the mean of the correspondent $\overline{m_i}$ is computed, and this is done for every N_i.
- *Phase 3:* The next step in the process is to replace each old value of m_i by its respective $\overline{m_i}$ value, and this replacement is done concurrently for all the m_i.

In our previous work, it has been shown that SOM is appropriate to classify automatically unfamiliar database entities[1]. Moreover, the SOM based visualization tool appeared to be for semantically identical or similar database entities exploration. This is due to SOM's inherit low dimensional regular grid layout.

B. SOM based Visualization

The most remarkable capability of SOM is that it produces a mapping of high-dimensional input space onto a low-dimensional (usually 2-D) map, where similar input data can be found on nearby regions of the map. Furthermore, SOM offers all the advantages of visual display for information retrieval which are: 1)the ability to convey a large amount of information in a limited space, 2) the facilitation of browsing and the perceptual inferences on retrieval interfaces, 3) the potential to reveal semantic relationship of documents [34]. These qualities will facilitate to the user the exploration of huge amount of database entities, and discover similar columns based on the semantic

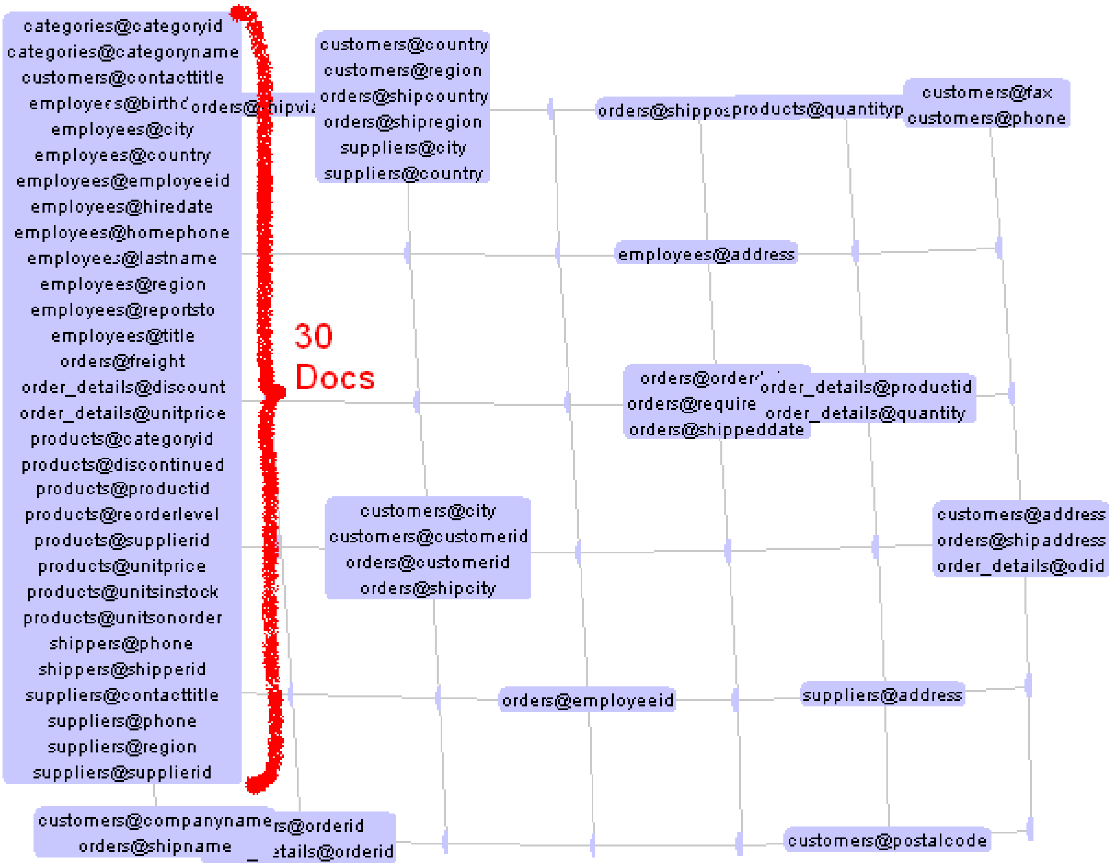


Fig. 2. Trained SOM map

content, which was not possible before with the traditional ontology based integration tools.

The easiest way to visualize the clustered documents (columns) is to match every column d_j to its respective Best Matching Unit (BMU) node[6] on the trained SOM's map. The BMUs are the SOM's map neurones. In essence, every document d_j is matched to its BMU by having the smallest Euclidean distance with it. This matching operation results in having semantically similar documents clustered on the same Map's node (BMU), let's call that cluster Cl_i. Even if the results were satisfactory, it has been observed, as shown in Fig. 2, that: (1) some nodes were too large because of grouping together multiple *heterogenous* clusters, and (2) some documents were not matched to their best class. That is to say, the map is unbalanced because of too many nodes are empty, while some other nodes are overloaded.

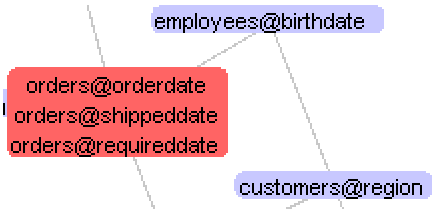


Fig. 3. Clusters Zooming: table@column

In addition, the SOM-based visualization tool offers different graphical features such as zooming, flipping the map, or enlarging the distance between the nodes. The zooming feature is shown in Fig. 3.

V. Post-Processing: CIBC

As a solution to the two previous issues, a new algorithm called *Common Itemset Based Classifier (CIBC)* is proposed in order to *smoothen* the clusters obtained in the previous section, and make the visual presentation clearer.

Firstly, CIBC refines the SOM nodes' clusters by validating the homogeneity of every cluster Cl_i, and re-clustering them into more homogeneous sub-clusters, when necessary. However, as shown on Fig. 4, CIBC try to preserve the original topology by trying to move the new sub-clusters within neighborhood. Secondly, the algorithm finds for every unclustered column a possible matching cluster Cl_i. Finally, it distinguishes visually on the map the clusters with homogenous documents from clusters (nodes) with heterogenous documents.

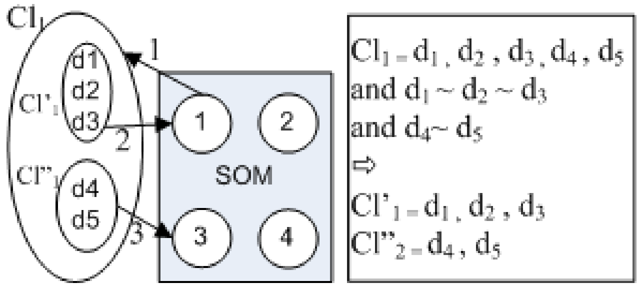


Fig. 4. Overview of CIBC

To illustrate the CIBC algorithm, suppose that we have a normalized term-document input VSM matrix, that has been pre-processed and then processed through the Batch version of SOM. The next step is to tune up the trained map using CIBC algorithm as follow:

A. Phase 1: Forming Itemsets

Let us assume that we are given a set of document items D. Suppose that an itemset I_{t_k}, where $I_{t_k} \subseteq D$, is some subset of *similar* documents d_j (at least 2) based on the common term t_k:

$$I_{t_k} = \{d_j \in D \mid w_{jk} > 0 \wedge |I_{t_k}| \geq 2\} \quad (1)$$

where, w_{jk} is an entry in the VSM matrix and it represents the weight of term j in the document k.

At this point, some itemsets contain a high number of similar documents because of stop words like "the" for example or some other type of noise. Therefore, in order to eliminate these insignificant itemsets, the ratio of similar documents in every itemset $|I_{t_k}|$ should be smaller than a certain threshold called Max_I:

$$\frac{|I_{t_k}|}{|D|} < Max_I (2),$$

where, $0 < Max_I < 1$

For example, we can fix Max_I to 0.7. It means that if an itemset is formed of 70% of the documents (entities) or more, then they should be eliminated because they have probably in common an insignificant term such as a stop word like "the, a, an". In fact, the purpose of the condition (2) is to reduce the number of potential itemsets to the essential ones. Besides, it permits to reduce the execution time of the post-processing algorithm.

However, one of the problems is that Max_I has to be proportionally increased to the dimensionally reduction ratio. In other words, the more the dimension is reduced, the more Max_I needs to be relaxed. First, there in no exact formulas yet to increase the Max_I parameter. Secondly, Max_I can not be increased indefinitely. Once the original VSM matrix is reduced approximately by 4, the Max_I needs to equal to almost 1 for better results. Consequently, Max_I looses it sense. This to say, the CIBC becomes limited to a reduction ratio of 1/4. For more details, refer to the experiments section.

B. Phase 2: SOM Nodes's Clusters Homogeneity Validation

As explained previously, some cluster Cl_i, obtained from the SOM's visualization phase, are not heterogeneous and as a consequence some nodes are overloaded by documents and many other ones are complectly empty. Therefore, the heterogeneity of every cluster Cl_i should be validated.

First, for every cluster Cl_i has to be found all the itemsets I_{t_k} having at least two documents in common with it. Then, it should be kept only the intersection of the two subsets $Cl_i \bigcap I_{t_k}$, named: I_{Cl_i,t_k} .

Let us call all the identical itemsets I_{Cl_i,t_k} : $I_{Cl_i,n}$, where $n \in \mathbb{N}$.

Secondly, among all the itemset $I_{Cl_i,n}$, it should be found the one having the *biggest* number of documents, and let us called it: Cl'_i. However, Cl'_i should respect the following condition in order to keep only the strongly semantically related documents:

$$\frac{|T_{Cl'_k}|}{|T_{Max(d_j,Cl'_k)}|} > \alpha, \; (3)$$

where, $|T_{Cl'_k}|$ is the size of the vocabulary of the sub-cluster Cl'_k, $|T_{Max(d_j,Cl'_k)}|$ is the vocabulary size of the document d_j having the richest vocabulary and which belongs to Cl'_k. And α (usually $= 0.05$) is a threshold to keep only strongly related documents of the current sub-cluster Cl'_k.

Another problem in the CIBC algorithm is that there no exact formulas to calculate the threshold α. Usually when equal to 0.05, it gives good results. However, sometimes this value varies between [0.02-0.075].

The sub-cluster Cl'_k is kept on its original node (BMU) while the remaining documents $\overline{Cl'_k}$ are reprocessed until no homogeneous sub-cluster can be found. In case where there are other existing sub-clusters $Cl”_k$, each one of them should be moved to a another empty node (with no clusters), ideally, within the neighborhood.

C. Phase 3: Clustering the Unclustered Documents

For every unclustered document, it should be found, respecting the rule (1), the best matching cluster if it exists. Otherwise, as last resort, by following the same process (phases 1 and 2), it should tried to form new clusters among the unclustered documents.

D. Phase 4: Re-mapping the Unclustered Documents

At this point, only left a certain number of semantically unique documents for which nor a matching cluster nor another similar document could be found. Therefore to show visually their uniqueness, these documents are reassigned to their first respective *empty* BMU. In the case where there is no available node, then they should be re-mapped to their first available BMU, which has only unclusterd documents. In the consequence of which, it will be formed new clusters of unclustered documents UCl_k.

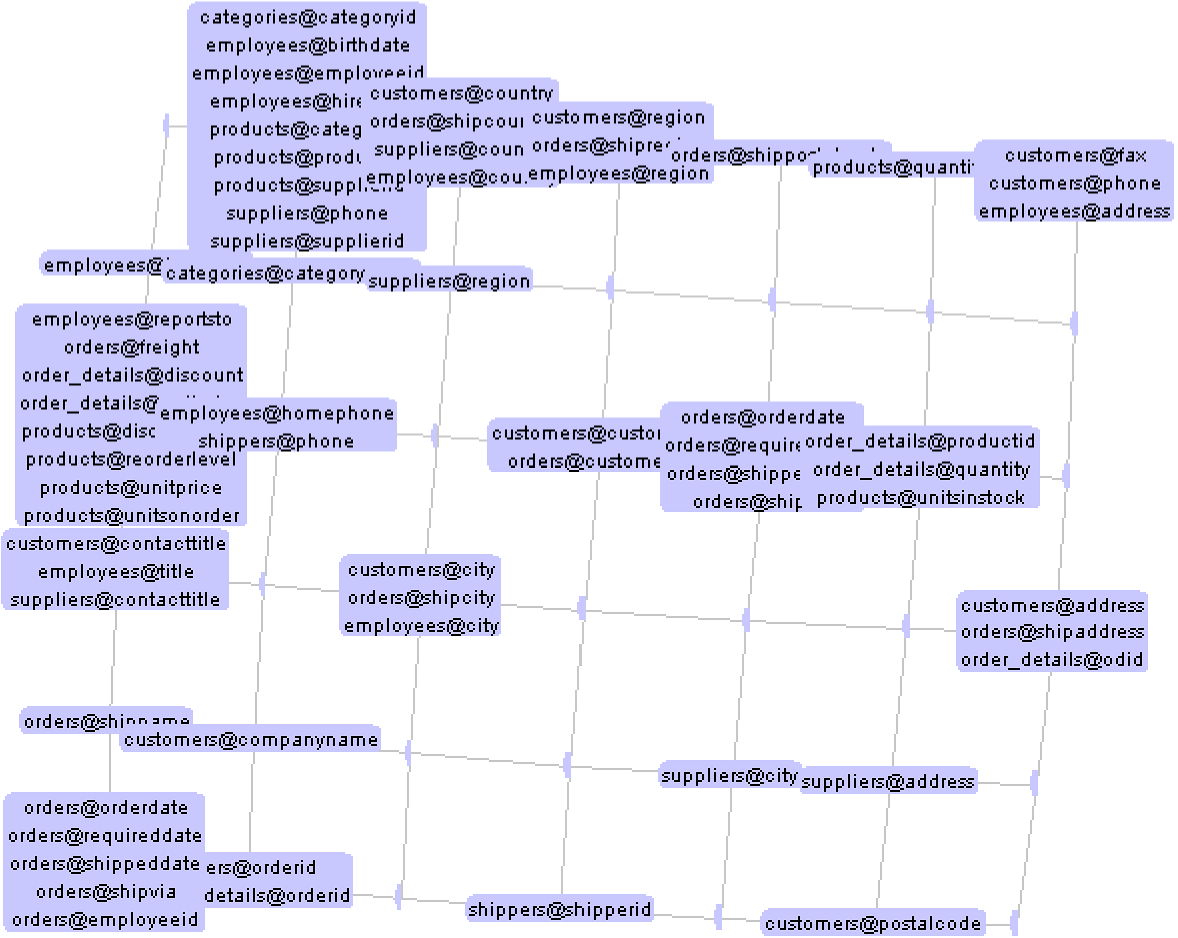


Fig. 5. SOM's Updated Map

E. Visualization Update

The SOM map is updated with the redistribution of homogenous clusters of columns, as well as the construction of clusters with unclustered documents (columns). As an illustration of an updated SOM map, the map shown on Fig. 5 (next page) is the original map shown on Fig. 3 (previous page) after applying the proposed CIBC algorithm to it. In the case where there is heterogenous clusters or group of heterogenous documents UCl$_k$, the clusters Cl$_k$ with homogenous documents, formed in phase 2 and 3, will be differentiated visually by distinguished colors.

VI. Experiments

The experiments are divided into two case studies. The first case study (Northwind) being the first work related to unified vectorization, evaluates textual and numerical SOM based data processing by unified vectorization(UV). Different traditional tokenization and vectorization techniques are evaluated. In the same case study, another series of tests serve to measure the RP technique and the proposed CIBC post-processing Algorithm. In the second case study, another much bigger database (Sakila) is used to evaluate unified vectorization on more realistic scale. The database having a larger portion of numerical data serves to assess the proposed BF-IDBF weighting measure. In addition, the experiments permit to compare the usage of CIBC versus the combination of TF-IDF and BF-IDBF on small and larger scales.

A. Corpus

The proposed techniques are tested on two demo databases available online, named: Northwind and Sakila. As shown on Tab. 2, the Northwind database has a 77 tables with 4681 terms as vocabulary size, which permits to test the algorithms with and without dimensionality reduction, and compare results. The Sakila Database is much bigger with 22104 term, and has more consistent volume of numerical data (17172) which permits to have more realistic database, and it's more appropriate for BF-IDBF evaluation.

Tab. 2. Unified Vectorization of Textual and Numerical Data

Data Set	Columns	terms	text	num. terms	Cat.
Northwind	77	4681	2154	2527	15
Sakila	89	22104	4932	17172	20

B. Tokenization

The process of breaking a text up into its constituent tokens is known as tokenization. Because we don't make any linguistic pretreatment; i.e. we do not need to apply lemmatization, stemming or stop words elimination; the impact of tokenization on the results increases. It is not the purpose of this paper to evaluate the impact of tokenization, however we consider important to mention some important facts. For example, when using bag of words as method of representation, if the non alphanumeric characters, such as ”()-+;.”, are not eliminated, the results could be affected. The F-measure can drop when using the SOM algorithm by up to 15%, and it is even worse when using the hybrid SOM and CIBC algorithm (- 25%). Therefore, all the non alphanumeric characters are eliminated for the experiments.

C. Evaluation Measures

One of the most used performance evaluation of unsupervised classifiers in the IR literature, with respect to the known classes for each document, are F-measure and Entropy which are based on Precision and Recall [24]:

$$P = Precision(i,j) = \frac{N_{ij}}{N_j}$$

$$R = Recall(i,j) = \frac{N_{ij}}{N_i}$$

where, N_{ij} is the number of members of the class i and the cluster j, N_j is the number of members of the cluster j, and N_i is the number of members of the class i.

F-measure distinguishes the correct classification of document labels within different classes. In essence, it assesses the effectiveness of the algorithm on a single class, and the higher it is, the better is the clustering. It's defined as follow:

$$F(i) = \frac{2PR}{P+R} \Longrightarrow F_c = \frac{\sum_i(|i| \times F(i))}{\sum_i |i|}$$

where; for every class i is associated cluster j which has the highest F-measure, F_c represents the overall F-measure that is the weighted average of the F-measure for each class i, $|i|$ is the size of the class i.

D. Configuration

The tests were conducted using different machines. The most recent ones were conducted on old machine: P. 4, 2.81 GHz, 1 Gig of RAM. Usually, the learning time is around 2 minutes. It does not change because the data is reduced always to the same dimensionality (2000 terms) using RP. Consequently, the learning time is not significant in this research paper, that is why the learning time was not measured for every test run.

E. Evaluation

The evaluation of the relevance of the classes formed remains an open problem because of the subjective nature of the task [24]. There are often various relevant groupings for the same data set. For instance, when comparing quantity entries (e.g. 12 05) with a date entry (e.g. 12-02/2005) the data are similar numerically but it does not fit within the purpose of this research which is the integration and the visualization of semantically similar columns. Therefore, these kind of data were classified in separated classes and our hybrid algorithm was significantly penalized (up to 25%) in this sense, but on the other hand it creates future research perspectives and challenges that are also closer to the industrial needs.

F. Case Study 1: Northwind

The objective of these tests is to evaluate, and at the same time compare, the performance of the algorithms of SOM with different weighting measures, including BF-IDBF versus the hybrid proposed algorithm (SOM with CIBC). Another experimental interest is to select the best text representations, among those proposed in section 2, in order to obtain the best clustering result for heterogeneous data mining. Then, the best data representation would be tested on larger data to evaluate the scalability of the proposed methods.

G. Preliminary tests: Without Dimension Reduction

A good classification requires a good presentation[24]. However, the vast number of text representation possibilities presented earlier requires to select the most relevant ones to continue further our tests. In this sense, firstly it will be tested on Northwind DB, without dimensionality reduction, different combination of tokenization and vectorization as show in the Tab. 2. Accordingly, there are two tokenization methods: bag of words versus N-gram described earlier in the section III. Besides, there are two text vectorization techniques, which are TF-IDF and binary. the binary tokenization is simple; if the term is in document then weight is equal to 1, otherwise the value is 0. Regarding the numerical vectorization techniques, there is only historgram for this test. However, it is important to mention that some tests consider the numerical terms as texts, e.g. the term ”195” would be treated as a text. In brief, there are three possible vectorization methods: TF-IDF (text) with histogram (numeric), binary(text) and histogram(numeric), and TF-IDF (text and numeric) where the numbers are considered as text terms.

Tab. 3. Preliminary F-measure with different representations

Data Set	Classifiers	tf-idf + hist.	bin. + hist.	tf-idf
SOM	Bag of words	58.24	49.85	54.37
Hybrid	Bag of words	**87.64**	74.18	65.07
SOM	Ngram	51.55	45.12	51.26
Hybrid	Ngram	59.75	52.34	60.18

The preliminary tests (Tab. 3) show an evident performance advance of the hybrid (SOM+CIBC) algorithm by unified vectorization over pure SOM; there is improvement of the quality of clustering by [7.22-29.4]% of the F-Measure. The best results of the hybrid algorithm are when using bag of words as tokenizer, and with the proposed combination of TF-IDF and histogram as vectorization method. However, it can be observed that N-gram (3-gram) tokenization does not improve the results when used with proposed unified vectorization. In brief, the hybrid algorithm(SOM+CIBC) performs better than the pure SOM. More important is the fact that the proposed SOM-based HDM by UV using works better than processing the data based on one data type.

H. Tests: With Dimensionality Reduction

Dimensionality reduction was applied for the test series of this section, which are resumed in table 4 and Fig. 6. The size of the textual data vocabulary for all vectorization type (all test scenarios) was originally 2154 terms, then the dimension was reduced to 1000 by using RP. The dimension of the numerical data was not reduced using RP because Northwind is a small database and histograms reduced it enough.

From the results, we can see that the best performance in general is the usage of the proposed Hybrid algorithm of SOM and CIBC. First of all, as shown on Tab.5, CIBC improves the classification precision of SOM by [4.84 - 23.78]% (F-Measure). Secondly, we can see that the proposed integration technique of numerical and textual data works very well, particularly with bags of words tokenization. Another interesting remark, is that the proposed unified vectorization technique (TF-IDF + histogram) performs better when using bag of words tokenization for textual data rather than N-Gram. This is valid for both processing algorithms.

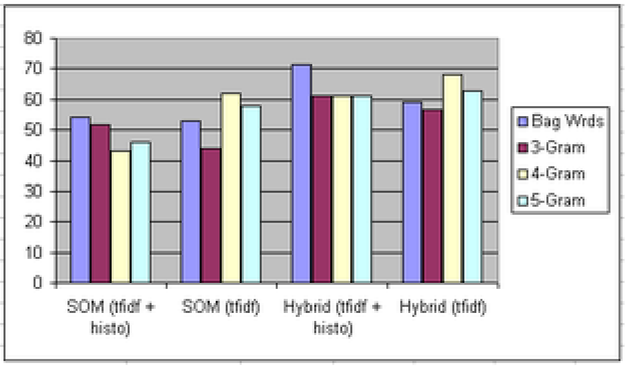


Fig. 6 F-measure comparison depending on tokenization, vectorization, and classification algorithms

We were expecting N-Gram to perform better that bagwords because usually N-gram performs better than bag of words. But, the most surprising is the fact that the combination of TF-IDF and BF-IDBF performs very similarly to TF-IDF and histogram. As it will shown in the next series of experiments, the duo of TF-IDF and BF-IDBF performs not only better that TF-IDF and histogram, but it performs even better than CIBC (TF-IDF and histogram). The only explanation that we can find to this lower F-Score is the fact that Northwind is not appropriate DB to evaluate BF-IDBF because of its low amount of numerical data (4681). In addition, the majority of these numbers, because the data is synthetic, are repeated dozen of times in the corpus, which probably lowers the BF-IDBF score. Consequently, we think that the Northwind data set is not appropriate to evaluate Numerical data vectorization measures.

I. Case Study II: Sakila

The objective of these tests is to evaluate the add value of the BF-IDBF measure on the processing of heterogenous data type, and more specifically using SOM classification method. In order to measure the contribution of the BF-IDBF weight to the processing phase, F-measure is used. It is used with references to defined classes documents; The SOM's resulting clusters are compared to the classes which are the ideal clustering results.

Several representations are tested; therefore, multiple combinations of tokenization and vectorization are tried for SOM based processing. Accordingly, two tokenization methods, for textual data, are used: bag of words versus N-gram as described earlier. Besides, three vectorization techniques are compared: TF-IDF (textual data), histogram (numerical data) and the proposed BF-IDBF(numerical data).

It's important to mention that around 60 % of the Sakila database files (columns) are constituted of purely numerical data such as keys, dates, prices, phone numbers, etc. The remaining ones are textual or a combination of the two data types. This to say that Sakila reassembles more to the real industrial databases because usually there are more numerical keys due to primary keys and other numerical informations. In addition, by having more numerical files, it permits to evaluate better the contribution of the BF-IDBF measure to the classification algorithm.

For every measure, four sets of tests were completed. Firstly, the database classification using SOM was evaluated without unified vectorization, i.e. either numerical or textual *exclusive* input data were processed. In this case, the dimension was reduced using RP to 1500. Then, the heterogenous textual and numerical data type were processed simultaneously by unified vectorization using different vectorization including BF-IDBF. Therefore, it was possible to estimate the enhancement obtained by the proposed measure. Note that in the second case, the dimension was reduced to 1250 for textual data and 750 for numerical data for a total dimension of 2000. In other words, there is a loss of information when the data is processed by unified vectorization, but we don't think it has a major impact to bias the results.

As illustrated on (Tab. 6, Fig.7), the experiments show that the proposed combination of TF-IDF and

Tab. 5 Comparison of the F-score values

	SOM			(SOM + CIBC)	
Tokenization	TF-IDF + Histogram	TF-IDF + BF-IDBF	TF-IDF	TF-IDF + Histogram	TF-IDF
Bag of words	54.03	54.74	52.93	**71.42**	59.21
3-Gram	51.63	49.91	43.83	66.66	56.47
4-Gram	42.88	50.85	62.02	66.66	68.28
5-Gram	46.00	45.39	58.04	61.00	62.88

Tab. 6 Precision measures with different representations

Tokenization	TF-IDF	TF-IDF+HISTO	(TF-IDF+HISTO)+CIBC	TF-IDF+BFIDBF
Bag of words	26.68	46.23	52.81	64.81
3-Gram	21.68	38.15	43.11	58.77
4-Gram	30.02	42.72	45.23	**65.56**
5-Gram	26.57	47.28	54.64	63.94

Tab. 7 Recall measures with different representations

Tokenization	TF-IDF	TF-IDF+HISTO	(TF-IDF+HISTO)+CIBC	TF-IDF+BFIDBF
Bag of words	89.89	74.15	81.58	86.52
3-Gram	**93.26**	70.79	82.02	86.52
4-Gram	92.13	71.91	71.05	**88.76**
5-Gram	92.13	71.91	71.05	83.14

Tab. 8 F-measures with different representations

Tokenization	TF-IDF	TF-IDF+HISTO	(TF-IDF+HISTO)+CIBC	TF-IDF+BFIDBF
Bag of words	41.15	56.95	64.11	74.11
3-Gram	35.18	49.58	56.51	69.99
4-Gram	45.29	53.59	55.28	**75.42**
5-Gram	41.25	57.05	61.77	72.29

BF-IDBF vectorization measures enhances the precision of SOM significantly by at least 15% when compared to the combination of TF-IDF and histogram. The precision is even better than applying the CIBC post processing algorithm. The best precision results are obtained when using 4-gram as tokenizer which improves the SOM's precision by almost 20 % . Furthermore, the precision results obtained using *exclusively* BF-IDBF (54.49%) were better than even the combination of TF-IDF and histogram.

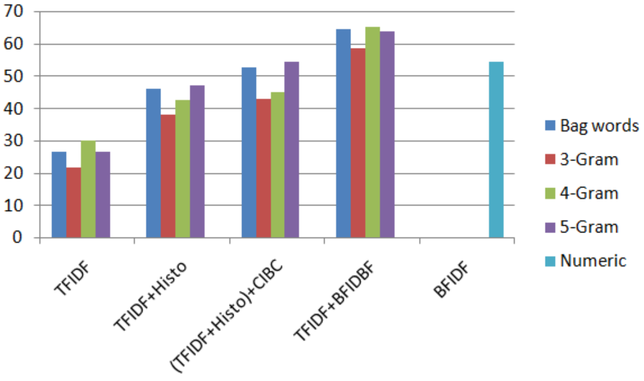


Fig. 7. Precision comparison depending on tokenization, vectorization and classification algorithm

In respect to the Recall measure (Tab. 7, Fig. 8), the best performance was with the exclusive usage of the TF-IDF vectorization measure; however, its precision was very low, and that is why F-measure is a more objective way of comparing these representations. Then, the proposed combination of TF-IDF and the new BF-IDBF vectorization measures follow in the second position. It is interesting to note that again the exclusive usage of the BF-IDBF measure with purely numerical data (69.66%) is almost as good as the unified vectorization by TF-IDF and histogram.

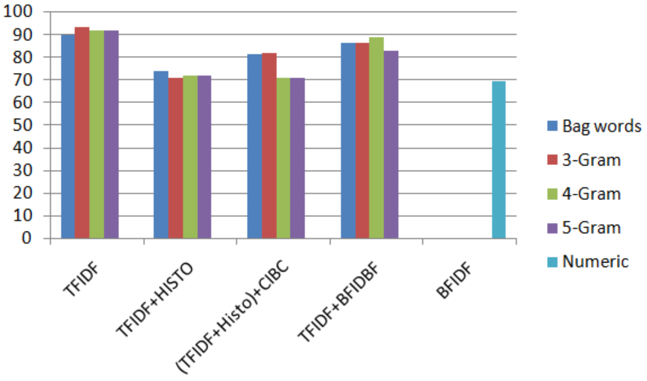


Fig. 8. Recall comparison depending on tokenization, vectorization and classification algorithm

Finally, the precision and the recall are combined equally to produce the F-measure which is more objective to compare the different representations. It can be easily observed (Tab. 8) that the 4-gram tokenization combined with the proposed vectorization using TF-IDF and BF-IDBF overcomes all the other representations.

In fact, it performs better than the unified vectorization by TF-IDF and histogram by around 20%, and it almost *doubles* the performance of the traditional textual representation by TF-IDF. Even the usage of the pure BF-IDBF representation of the exclusively numeric data(61.15%) performs better than the pure TF-IDF or even the combination of TF-IDF and histogram. Furthermore, even the post-processing algorithm CIBC is applied to the combination of TF-IDF and histogram, the proposed combination of TF-IDF and BF-IDBF show better results by at least 10%. This demonstrates the beneficial properties of the proposed BF-IDBF measure for heterogeneous data mining by unified vectorization on large data sets. Generally, it appears that similar data representations, such as TF-IDF and BF-IDBF, are more appropriate for heterogeneous data mining by unified vectorization. In fact, as illustrated on Fig. 9, the impact of the vectorization measures is more significant than the impact of the tokenization measures on the clustering results. In addition, the Fig. 9 illustrates well the importance of the pre-processing phase as one of the most important phases in data classification because of the major impact that it can have on the machine learning clustering results. Finally, per induction we can expect the proposed combination of TF-IDF and the new BF-IDBF measure to be possibly applied to any other machine learning algorithm for better heterogeneous data mining results.

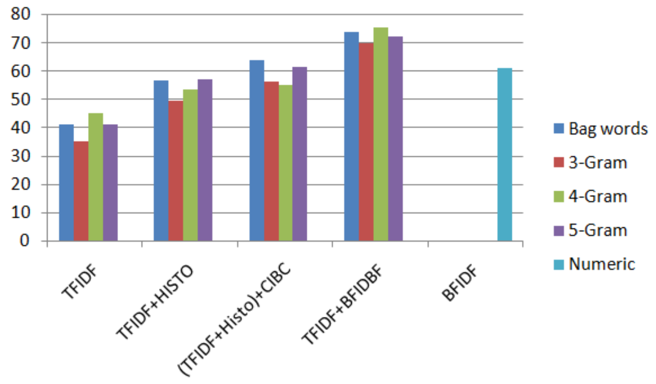


Fig. 9. F-measure comparison depending on tokenization, vectorization and classification algorithm

VII. Conclusions

In this paper we have presented an efficient way to process heterogenous textual and numerical data, for data integration purposes, by the usage of the SOM-based visualization tool. By focusing on the pre-processing and post-processing phases, we demonstrated using SOM based processing by unified vectorization, that it is possible to extract convergent clustering results from heterogeneous textual and numerical data types despite the heterogeneity of the data type. The SOM visualization tool exposes the similarity between database columns based on their semantical content, which greatly serves the purpose of distributed database integration. This tool is applicable to data integration over web data sources. To evaluate the best configuration of the SOM-based tool, we have compared several pre-processing methods, additionally to the CIBC post-processing method, for heterogeneous data mining by unified vectorization using SOM.

First, we tried several text tokenization (Bag of words, 3-Gram, 4-Gram, 5-Gram). Even if their impact is minor on heterogeneous data mining clustering results, nevertheless 4-Gram shows generally the best performances.

Secondly, we tried several vectorization measures(text: TF-IDF, numeric:histograms and BF-IDBF), which may have much more important impact on heterogeneous data mining clustering results. The results differ depending on the size of the databases. With smaller databases, which have smaller data corpus and require a partial dimensional reduction ratio (maximum 1/4), the combination of TF-IDF and histogram versus TF-IDF and the proposed BF-IDBF have very similar results. However, the couple (TF-IDF, histogram) combined with the CIBC offer better results than the couple (TF-IDF, BF-IDBF). This is true on small databases only because once the database is large enough to require a dimension reduction ratio superior to 1/4, the performances of CIBC and histograms are penalized. In fact, at that point CIBC method offers a limited improvement which make suitable for small databases only. In the opposite, with larger data sets that are more similar to industrial database, the couple (TF-IDF, BF-IDBF) offer a clear amelioration of heterogeneous data mining results. Even when compared to the couple (TF-IDF, histogram) combined with the CIBC algorithm, the couple (TF-IDF, BF-IDBF) is better. Consequently, the combination (TF-IDF, BF-IDBF) is more suitable because it offers better results, and it is faster because it does not require the usage of a post-processing algorithm such as CIBC. Generally speaking, it seems that the usage of similar vectorization methods, such as TF-IDF and BF-IDBF for example, lead to better heterogeneous data mining results.

In our future work we want to apply the proposed methods to other domains such as medical, finance, or network intrusion detection fields. Furthermore, we are considering to improve the pre-processing techniques for better heterogeneous data mining by unified vectorization. In addition, we would like to apply the unified vectorization method with other processing techniques. Finally, we aim to integrate other data types such as images, dna, and others.

Acknowledgements

The authors wish to thank the anonymous reviewers for their valuable suggestions.

References

[1] Bourennani, F., Pu, K. Q., Zhu, Y., Visualization and Integration of Databases using Self Organizing Maps, Proceedings of the International Conference on Advances in Databases, Knowledge, and Data Applications (DBKDA09), Cancun, Mexico, 2009, pp. 155-160.

[2] Bourennani, F., Pu, K. Q., Zhu, Y. Visual Integration Tool for Heterogeneous Data Type by Unified Vectorization. Proceedings of the 10th IEEE International Conference in Reuse and Integration (IRI'09), Las-Vegas, USA, 2009, pp. 132-137.

[3] Colomb, R. M. Impact of Semantic Heterogeneity on Federating Databases. The Computer Journal, Vol. 40, no 5, 1997, pp. 235-244.

[4] Rahm, E. and Bernsteinp, P. A. A survey of approaches to automatic schema matching. The VLDB Journal, Springer-Verlag New York, Inc., Secaucus, Vol. 10, no 4, NJ, USA, 2001, pp. 334-350.

[5] Robertson, G. G., Czerwinski, M. P., Churchill, J. E. Visualization of Mappings Between Schemas. ACM SIGCHI Conference on Human Factors in Computing Systems, Portland, Oregon, USA, 2005, pp. 431-439.

[6] Kohonen, T. Self-Organizing Maps. Berlin : Springer-Verlag, 2001.

[7] Lenzerini, M. Data Integration: A Theoretical Perspective. PODS'02, 2002. pp, 233-246.

[8] Shvaiko, P., and Euzenat, J., A Survey of Schema-Based Matching Approaches. Journal on Data Semantics, Vol. 4, 2007, pp. 146-171.

[9] Noy, N. F. Semantic Integration: A Survey Of Ontology-Based Approaches. SIGMOD Record, Vol. 33, no 4, 2004.

[10] Miller, R., Haas, L. M., A. Hernandez, M. Schema Mapping as Query Discovery. VLDB '00: Proceedings of the 26th International Conference on Very Large Data Bases, San Francisco, CA, USA, 2000, pp. 77-88.

[11] Wache, H., Veogele, T., Visser, U., Stuckenschmidt, H., Schuster, G., Neumann, H. and Hubner, S. Ontology-Based Integration of Information - A Survey of Existing Approaches. The Workshop on Ontologies and Information Sharing at the International Joint Conference on Artificial Intelligence (IJCAI), Victoria, BC, Canada, 2001. pp. 108-117.

[12] Salton, G. Automatic Text Processing. MA : Addison-Wesley, 1989.

[13] Van Rijsbergen, C.J., Information Retrieval 2nd ed.: Butterworth-Heinemann, 1979.

[14] Sheth, A. P. Changing Focus on Interoperability in Information Systems: From System, Syntax, Structure to Semantics. Norwell, Massachussetts, USA : M. F. Goodchild, M. J. Egenhofer, R. Fegeas, and C. A. Kottman (eds.) Kluwer, Academic Publishers, 1998, pp. 5-30.

[15] Kloptchenko, A., Eklund, T., Karlsson, J., Back, B., Vanharanta, H., Visa, A. Combining data and text mining techniques for analysing financial reports. Intelligent Systems in Accounting Finance and Management, Vol. 12, no 1, 2004. pp. 29 - 41.

[16] Back, B., Toivonen, J., Vanharanta, H., Visa, A. Comparing numerical data and text information from annual reports using self-organizing maps. International Journal of Accounting Information Systems, Vol. 2, no 4, 2001. pp. 249-269.

[17] Eklund, T., Back, B., Vanharanta, H., Visa, A. Benchmarking International Pulp and Paper Companies Using Self-Organizing Maps. Turku, Finland : TUCS Technical Report No 396, Turku Centre for Computer Science, 2001.

[18] Hearst, M. A. Untangling Text Data Mining. Proceedings of the 37th annual meeting of the Association for Computational Linguistics on Computational Linguistics, College Park, Maryland, USA, 1999. pp. 3-10.

[19] Rbov, I., Konecn, V., Matiov, A. Decision Making with Support of Artificial Intelligence. Agricultural Economics, Vol. 51, no 9, 2005, pp. 385-388.

[20] Parvizian, J., Tarkesh, H., Farid, S., Atighehchian, A. Project Management Using Self-Organizing Maps. Industrial Engineering and Management Systems, the official journal of APIEMS, Vol.5, no 1, 2006.

[21] Miller, R., Haas, L. M., A. Hernandez, M. Schema Mapping as Query Discovery. VLDB '00: Proceedings of the 26th International Conference on Very Large Data Bases, San Francisco, CA, USA, 2000, pp. 77-88.

[22] Baeza-Yates, R. and Ribeiro-Neto, R., Modern Information Retrieval. : Addison Wesley Longman, 1999.

[23] K. Lagus, S. Kaski, and T. Kohonen. Mining massive document collections by the WEBSOM method. Information Sciences, Vol.163, 2004. pp. 135-156.

[24] Amine, A., Elberrichi, Z., Bellatreche, L., Si- Monet, M., Malki, M. Concept-based clustering of textual documents using SOM. In Proceedings of the IEEE/ACS International Conference on Computer Systems and Applications, Doha, Quatar. 2008.

[25] Y. Miao, V. Keelj, and E. Milios. Document Clustering Using Character N-Grams:A Comparative Evaluation With Term-Based and Word-Based Clustering. 14th ACM International Conference on Information and Knowledge Management, Bremen, Germany, 2005.

[26] Sahami, M. Using Machine Learning to Improve Information Access. PhD thesis, Computer Science Department, Stanford University, 1999.

[27] Sebastiani, F. Machine learning in automated text categorization. ACM Computing Surveys, Vol. 34, no 1, 2002. pp. 1-47.

[28] Fradkin, D., Madigan, D. Experiments with Random Projections for Machine Learning. Ninth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, Washington, D.C, USA, 2003, pp. 517 - 522.

[29] Bingham, E. and Mannila, H. Random projection in dimensionality reduction: Applications to image and text data. Seventh ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, San Francisco, USA, 2001, pp. 245 - 250.

[30] Han, J., Kamber, M. Data Mining, Second Edition: Concepts and Techniques. San Francisco : Morgan Kaufmann, 2006. pp. 72-97.

[31] Magnello, M. E., Karl Pearson and the Origins of Modern Statistics: An Elastician becomes a Statistician, The New Zealand Journal for the History and Philosophy of Science and Technology, Vol. 1.

[32] Pyle, D. Data Preparation for Data Mining. San Francisco : Morgan Kaufman Publishers, 1999.

[33] Lagus, K. Text Mining with the WEBSOM, PhD thesis, Department of Computer Science and Engineering, Helsinki University of Technology, 2000.

[34] Lin, X. Map displays for information retrieval. Journal of the American Society for information Science, Vol. 48, no 1, 1997, pp. 40-54.

[35] Carpenter, G. A and Grossberg, S. ART2: Self-Organization of Stable Category Recognition Codes for Analog Input Patterns. Vol. 26, no 23, 1987, pp. 4919-4930.

[36] Alseshunas, J.J, St. Clair, D. C., Bond, W.E. Classification Characteristics of SOM and ART2, In Proceedings of the 1994 ACM symposium on Applied computing, Phoenix, USA, 1994, pp. 297 - 302.

[37] Song, M., Wu, YF (eds.). Handbook of Research on Text and Web Mining Technologies. USA : Idea Group Inc., 2008.

Live Geography – Embedded Sensing for Standardised Urban Environmental Monitoring

Bernd Resch [1,2]
Research Scientist
berno
[at] mit.edu
Member, IEEE

Manfred Mittlboeck [1]
Key Researcher
manfred.mittlboeck
[at] researchstudio.at

Fabien Girardin [2,3]
Researcher
fabien.girardin
[at] upf.edu

Rex Britter [2]
Visiting Professor
rb11
[at] eng.cam.ac.uk

Carlo Ratti [2]
Director and
Associate Professor
ratti
[at] mit.edu

[1] Research Studios Austria
studio *i*SPACE
Leopoldskronstrasse 30
5020 Salzburg, Austria

[2] MIT
SENSEable City Lab
77 Massachusetts Avenue
building 10, room 400
Cambridge, MA 02139, USA

[3] Universitat Pompeu Fabra
Department of Information and
Communication Technologies
Passeig de Circumval.lació 8
08003 Barcelona, Spain

***Abstract* – Environmental monitoring faces a variety of complex technical and socio-political challenges, particularly in the urban context. Data sources may be available, but mostly not combinable because of lacking interoperability and deficient coordination due to monolithic and closed data infrastructures. In this work we present the *Live Geography* approach that seeks to tackle these challenges with an open sensing infrastructure for monitoring applications. Our system makes extensive use of open (geospatial) standards throughout the entire process chain – from sensor data integration to analysis, Complex Event Processing (CEP), alerting, and finally visualisation. We discuss the implemented modules as well as the overall created infrastructure as a whole. Finally, we show how the methodology can influence the city and its inhabitants by „making the abstract real", in other words how pervasive environmental monitoring systems can change urban social interactions, and which issues are related to establishing such systems.**

Keywords* – *Urban environmental monitoring; Standardised infrastructure; Real-time GIS data analysis; Situational awareness; Embedded sensor device.

I. INTRODUCTION

Environmental monitoring is a critical process in cities to ensure public safety including the state of the national infrastructure, to set up continuous information services and to provide input for spatial decision support systems. However, setting up an overarching monitoring system is not trivial. Currently, different authorities with heterogeneous interests each implement their own monolithic infrastructures to achieve very specific goals, as stated in our previous work [1]. For instance, regional governments measure water levels for flood water prediction, while local governments monitor air quality to dynamically adapt traffic conditions, and energy providers assess water flow in order to estimate energy potentials.

The fact that these systems tend to be deployed in an isolated and uncoordinated way means that the automatic assembly and analysis of these diverse data streams is impossible. However, making use of all available data sources is a prerequisite for holistic and successful environmental monitoring for broad decision support in an urban context. This applies to emergency situations as well as to continuously monitoring urban parameters.

One way to overcome this issue is the extensive use of open standards and Geographic Information System (GIS) web services for structuring and managing these heterogeneous data. Here, the main challenge is the distributed processing of vast amounts of sensor data in real-time, as the widespread availability of sensor data with high spatial and temporal resolution will increase dramatically with rapidly decreasing prices [2], particularly if costs are driven down by mass utilisation.

From a political and legal standpoint, national and international legislative bodies are called upon to foster the introduction of open standards in public institutions. Strong early efforts in this direction have been made by the European Union (EU) through targeted directives (s. chapter IV). These regulations support the development of ubiquitous and generically applicable real-time data integration mechanisms. Shifting development away from proprietary single-purpose implementations towards interoperable analysis systems will not only enable live assessment of the urban environment, but also lead to a new perception of the city by its inhabitants. Consequently, this may in turn foster the creation of innovative applications that treat the city as an interactive sensing platform, such as *WikiCity* [3], involving the people themselves into re-shaping the urban context.

This paper begins with a review of related work in several research areas. Then, challenges of environmental monitoring with particular respect to the urban context are elucidated, before we summarise the current legal frameworks for environmental data management. Thereafter, our *Live Geography* approach is presented, which aims to integrate live sensor measurements with archived data sources on the server side in a highly flexible and interoperable infrastructure. Finally, we present our thoughts on how environmental sensing and Geographic Information (GI) processing can affect the city and its inhabitants. The ultimate goal of this paper is to present our approach's potential impact on urban policy and decision-making, and to point out its portability to other application domains.

II. Related Work

The Live Geography approach is manifold in terms of both concepts and employed technologies. As such, there are several research initiatives that form part of the overall methodology. These are described below.

The first domain is **sensor network development for environmental monitoring**. The Oklahoma City Micronet [4] is a network of 40 automated environmental monitoring stations across the Oklahoma City metropolitan area. The network consists of 4 Oklahoma Mesonet stations and 36 sites mounted on traffic signals. At each traffic signal site, atmospheric conditions are measured and transmitted every minute to a central facility. The Oklahoma Climatological Survey receives the observations, verifies the quality of the data and provides the data to Oklahoma City Micronet partners and customers. One major shortcoming of the system is that it is a much specialised implementation not using open standards or aiming at portability. The same applies to CORIE [5], which is a pilot environmental observation and forecasting system (EOFS) for the Columbia River. It integrates a real-time sensor network, a data management system and advanced numerical models.

Secondly, there are a number of approaches to **leveraging sensor information in GIS applications**. [6] presents the SenseWeb project, which aims to establish a Wikipedia-like sensor platform. The project seeks to allow users to include their own sensors in the system and thus leverage the „community effect“, building a dense network of sensors by aggregating existing and newly deployed sensors within the SenseWeb application. Although the authors discuss data transformation issues, data fusion, and simple GIS analysis, the system architecture is not based on open (geospatial) standards, only standard web services. The web portal implementation, called SensorMap, uses the Sensor Description Markup Language (SDML), an application-specific dialect of the Open Geospatial Consortium (OGC) SensorML standard.

In [7], the author presents a sensing infrastructure that attempts to combine sensor systems and GIS-based visualisation technologies. The sensing devices, which measure rock temperature at ten minute intervals, focuses on optimising resource usage, including data aggregation, power consumption, and communication within the sensor network. In its current implementation, the infrastructure does not account for geospatial standards in sensor observations. The visualisation component uses a number of open standards (OGC Web Map Service [WMS], Web Feature Service [WFS]) and open-source services (UMN Map Server, Mapbender).

Another sensing infrastructure is described in [8]. The CitySense project uses an urban sensor network to measure environmental parameters and is thus the data source for further data analysis. The project focuses on the development of a city-wide sensing system using an optimised network infrastructure. An important parallel with the work presented in this paper is that CitySense also considers the requirements of sensor network setup in an urban environment.

A GIS mashup for environmental data visualisation is presented in the nowCOAST application [9]. Data from several public providers are integrated in a web-based graphical user interface. nowCOAST visualises several types of raw environmental parameters and also offers a 24-hour sea surface temperature interpolation plot.

The most striking shortcoming of the approaches described above and other related efforts is that their system architectures are at best partly based on open (geospatial) standards.

The third related research area is **real-time data integration for GIS** analysis systems. Most current approaches use web services based on the classic request/response model. Although partly using open GIS standards, they are often unsuitable for the real-time integration of large volumes of data. [10] establishes a real-time spatial data infrastructure (SDI), which performs several application-specific steps (coordinate transformation, spatial data generalisation, query processing or map rendering and adaptation), but accounts neither for event-based push mechanisms nor for the integration of sensor data.

Other approaches for real-time data integration rely on the costly step of creating a temporal database. Oracle's system, presented in [11], is essentially a middleware between (web) services and a continuously updated database layer. Like Sybase's method [12], the Oracle approach detects database events in order to trigger analytical actions accordingly. In [13], a more dynamic method of data integration and fusion is presented using on-the-fly object matching and metadata repositories to create a flexible data integration environment.

The fourth comprised research field is the development of an **open data integration system architecture** in a non-application-specific infrastructure. Recent research efforts focus on general concepts in systems architecture development and data integration, but there are mostly no concrete conclusions as to how to establish such an infrastructure. A more technical approach for ad-hoc sensor networks is described in [14], where the authors discuss application-motivated challenges to combining heterogeneous sensor measurements through highly flexible middleware components. The method is strongly application-motivated and thus very well-thought-out as far as specific implementation details are concerned.

III. Challenges of Urban Environmental Monitoring and Sensing

The urban context poses many challenges to environmental monitoring: not only are there significant technical and technological issues, but also social and political ones as well.

The key technological challenge is the integration of different data sources owned by governmental institutions, public bodies, energy providers and private sensor network operators. This problem can be tackled with self-contained and well-conceived data encapsulation standards – independent of specific applications – and enforced by legal entities, as discussed in chapter IV. However, the adaptation of existing sensors to new standards is costly for data owners and network operators in the short term, and so increased awareness of the benefits of open standards is required.

From a technical viewpoint, unresolved research challenges for ubiquitous urban monitoring infrastructures are manifold and include: finding a uniform representation method for measurement values, optimising data routing algorithms in multi-hop networks, and developing optimal data visualisation and presentation methods. The last is an essential aspect of decision support systems, as different user groups might need different views of the underlying information. For example, in emergency local authorities might want a socio-economic picture of the affected areas, while first-response forces are interested in topography and people's current locations, and the public might want general information about the predicted development of a disaster.

From a more contextual standpoint, an important peculiarity of the urban context is that there are large variations within continuous physical phenomena over small spatial and temporal scales. For instance, due to topographical, physical or optical irregularities, pollutant concentration can differ considerably, even on opposite sides of the street. This variability tends to make individual point measurements less likely to be representative of the system as a whole. The consequence of this dilemma is an evolving argument for environmental regulations based on comprehensive monitoring data rather than mathematical modelling, and this demand is likely to grow. Consequently, the deployment of many sensors allows for more representative results together with an understanding of temporal and spatial variability.

One way to overcome this issue is to „sense people" and their immediate surroundings using everyday devices such as mobile phones or cameras. These can replace – or at least complement – the extensive deployment of specialised city-wide sensor networks. The basic trade-off of this people-centric approach is between cost efficiency and real-time fidelity. We believe that the idea of using existing devices to sense the city is crucial, but that it requires more research on sensing accuracy, data accessibility and privacy, location precision, and interoperability in terms of data and exchange formats. Furthermore, measurements are only available in a quasi-continuous distribution due to the high spatial and temporal variability of ad-hoc data collection. Addressing this issue will require complex distribution models and efficient resource discovery mechanisms in order to ensure adaptability to rapidly changing conditions.

Another central issue in deploying sensor networks in the city is the impact of fine-grained urban monitoring, as terms like „air quality" or „pollutant dispersion" are only a surrogate for a much wider and more direct influence on people, such as life expectation, respiratory diseases or quality of life. This raises the demand of finding the right level of information provision. More accurate, finer-grained or more complete information might in many cases not necessarily be worthwhile having, as this could allow for drawing conclusions on a very small scale, in extreme cases even on the individual. This again could entail a dramatic impact in a very wide range of areas like health care, the insurance sector, housing markets or urban planning and management.

Finally, some more unpredictable challenges posed by the dynamic and volatile physical environment in the city are radical weather conditions, malfunctioning hardware, connectivity, or even theft and vandalism.

IV. Policy-Frameworks for the Integration of Real-time Sensor Information

As mentioned above, we have seen an explosion of spatial data collection and availability in digital form in the past several years. There are various national and international efforts to establish spatial data infrastructures (SDI) for promoting and sharing geospatial information throughout governments, public and private organisations and the academic community. It is a substantial challenge solving the political, technological and semantic issues for sharing geographic information to support decision making in an increasingly environment-oriented world. In 2007, the United Nations Geographic Information Working Group published a report subsuming recent regional national and international technologies, policies, criteria, standards and people necessary to organise and share geographic information. These include real-time location aware sensor measurements to develop a United Nations Spatial Data Infrastructure (UNSDI) and encourage interoperability across jurisdictions and between UN member states. As described, these SDIs should help stimulate the sharing and re-use of expensive geographic information in several ways:

- The *Global Spatial Data Infrastructure Association* is one of the first organisations to promote international cooperation in developing and establishing local, national and international SDIs through interaction between organisations and technologies supported by the U.S. Geological Survey (USGS).
- On a supra-national level, the *INfrastructure for SPatial Information in Europe* (INSPIRE) aims to enable the discovery and usage of data for analysing and solving environmental problems by overcoming key barriers such as inconsistency in data collection, a lack of documentation, and incompatibility between legal and geographic information systems.

- *Global Monitoring for Environment and Security* (GMES) is another European Initiative for the implementation of information services dealing with environmental and security issues using earth information and in-situ data for the short, mid and long-term monitoring of environmental changes. This is Europe's main contribution to the Group of Earth Observations (GEO) for monitoring and management of planet earth.
- *Global Earth Observation System of Systems* (GEOSS) seeks to establish an overarching system on top of national and supra-national infrastructures to provide comprehensive and coordinated earth observation for transforming these data into vital information for society.

The common goal of all these initiatives – and the numerous national SDI approaches – is the integration and sharing of environmental information comprising remote sensing data, geographic information and (real-time) measurement data sets. With the European Shared Environmental Information System (SEIS), a new concept has been introduced to collect, analyse and distribute these information sets in a loosely coupled, „federal“ structure focused on defining interfaces. A particular focus is dedicated to near real-time datasets like sensor measurements. This new flexibility should foster the integration of sensor measurements into existing SDIs.

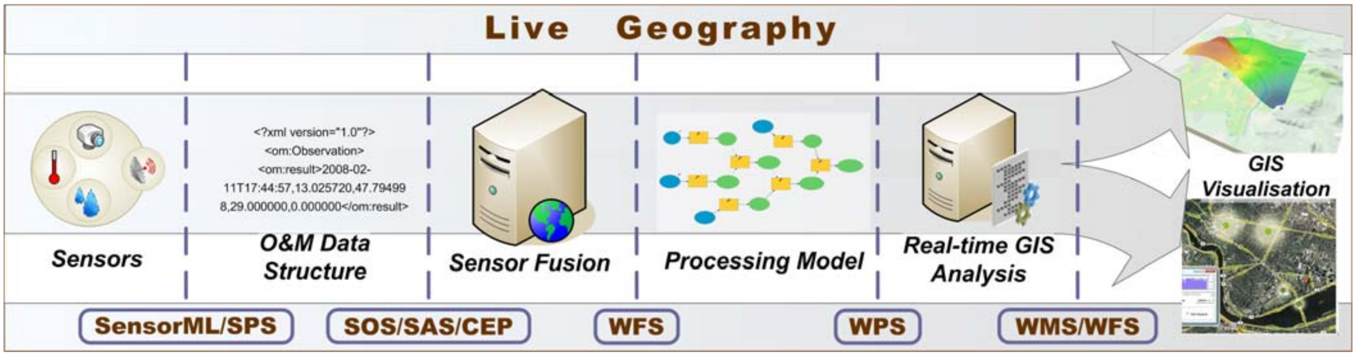
V. Live Geography Approach

With the above mentioned challenges to urban monitoring and standardisation in mind, we have created the *Live Geography* approach, which aims to combine live measurement data with historic data sources in an open standards-based infrastructure using server-side processing mechanisms.

The system architecture is composed of layers of loosely-coupled and service-oriented building blocks, as described in the following sections. In this way, data integration can be decoupled from the analysis and visualisation components, allowing for flexible and dynamic service chaining. In order to fulfil real-time data needs and alerting requirements, the concept also incorporates an event-based push mechanism (sub-section B). As noted above, one of the major challenges is the integration of location-enabled real-time measurement data into GIS service environments to perform distributed analysis tasks. There are three main requirements for quality-aware GIS analysis: *accuracy, completeness* and *topicality* of the input data (layers). As recent developments often do not account for time stamp parameters, it is necessary to identify effective ways to combine space and terrestrial real-time observation data with SDI information layers.

Fig. 1 illustrates the basic service infrastructure of the Live Geography concept. The general workflow within the infrastructure can be followed from left to right. First, heterogeneous data sources, such as sensor data, external data (provided via standardised interfaces such as OGC WFS or WCS [Web Coverage Service]) or archived data are integrated on the server side. This integration can happen via both classical request/response models and push services that send out alerts, e.g. if a certain threshold is exceeded. The flexible sensor fusion mechanism supports as well mobile as static sensors. Next, the different kinds of data are combined by a data integration server. This step requires real-time processing capabilities, such as Event Stream Processing (ESP) and Complex Event Processing (CEP). The harmonised data are then fed to pre-defined GIS process models to generate user-specific output.

This approach shifts resource-consuming geo-processing operations away from the client by executing complex, asynchronous analysis tasks on the server side, and then simply providing the client with a tailored result. The output could be an XML structure, a numerical value, or a contextual map tailored to the user's specific needs. The crucial benefit of this approach is that GIS applications – that previously offered GIS functionality only through resource-consuming desktop clients – could be replaced by lightweight web-based analysis tools. This enables the results of GIS analysis to be delivered to a wide variety of internet-connected devices, including personal computers, handhelds, and smart phones, or even other online analytical processes. This also allows for real-time situational awareness in spatial decision support systems. In other words, the system is suitable for using GIS-compliant data sets to assess urban environmental conditions and predict, within limits, their development in real-time. [15]



```
<?xml version="1.0"?>
<om:Observation>
<om:result>2008-02-
11T17:44:57,13.025720,47.79499
8,29.000000,0.000000</om:result>
```

Figure 1. Live Geography Infrastructure.

A. Usage of Open Standards

The components of this process chain are separated by several interfaces, which are defined using open standards. The first central group of standards is subsumed under the term Sensor Web Enablement (SWE), an initiative by the OGC that aims to make sensors discoverable, query-able, and controllable over the Internet [16]. Currently, the SWE family consists of seven standards, which encompass the entire process chain from making sensors discoverable in a registry, to measuring physical phenomena, and sending out alerts. [17]

- *Sensor Model Language (SensorML)* – This standard provides an XML schema for defining the geometric, dynamic and observational characteristics of a sensor. Thus, SensorML assists in the discovery of different types of sensors, and supports the processing and analysis of the retrieved data, as well as the geo-location and tasking of sensors.
- *Observations & Measurements (O&M)* – O&M provides a description of sensor observations in the form of general models and XML encodings. This framework labels several terms for the measurements themselves as well as for the relationship between them. Measurement results are expressed as quantities, categories, temporal or geometrical values as well as arrays or composites of these.
- *Transducer Model Language (TML)* – Generally speaking, TML can be understood as O&M's pendant or streaming data by providing a method and message format describing how to interpret raw transducer data.
- *Sensor Observation Service (SOS)* – SOS provides a standardised web service interface allowing access to sensor observations and platform descriptions.
- *Sensor Planning Service (SPS)* – SPS offers an interface for planning an observation query. In effect, the service performs a feasibility check during the set up of a request for data from several sensors.
- *Sensor Alert Service (SAS)* – SAS can be seen as an event-processing engine whose purpose is to identify pre-defined events such as the particularities of sensor measurements, and then generate and send alerts in a standardised protocol format.
- *Web Notification Service (WNS)* – The Web Notification Service is responsible for delivering generated alerts to end-users by E-mail, over HTTP, or via SMS. Moreover, the standard provides an open interface for services, through which a client may exchange asynchronous messages with one or more other services.

Furthermore, *Sensor Web Registries* play an important role in sensor network infrastructures. However, they are not decidedly part of SWE yet, as the legacy OGC Catalogue Service for Web (CSW) is used. The registry serves to maintain metadata about sensors and their observations. In short, it contains information including sensor location, which phenomena they measure, and whether they are static or mobile. Currently, the OGC is pursuing a harmonisation approach to integrate the existing CSW into SWE by building profiles in ebRIM/ebXML (e-business Registry Information Model).

An important ongoing effort in SWE development is the establishment of a central *SWE Common* specification. Its goal is to optimise redundancy and maximise reusability by grouping common elements for several standards under one central specification.

The functional connections between the described standards are illustrated in Fig. 2.

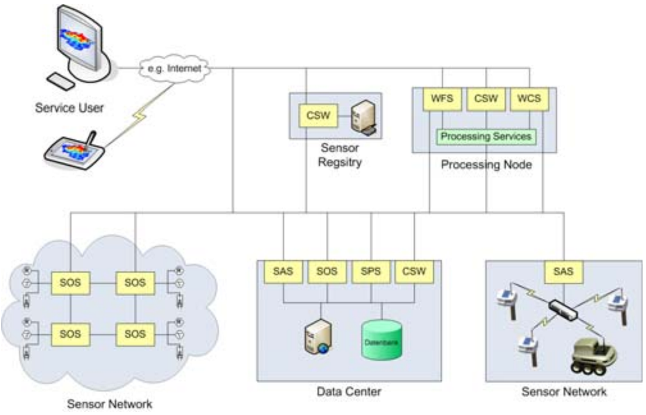


Figure 2. Functional Connections between the SWE Standards.

Besides these sensor-related standards, other OGC standards are used for data analysis and provisioning. The Web Processing Service, as described in [18], provides an interface to access a processing service offering a number of pre-defined analytical operations – these can be algorithms, simple calculations, or more complex models, which operate on geospatial data. Both vector and raster data can be processed. The output of the processes can be either a pre-defined data structure such as Geographic Markup Language (GML), geoRSS, Keyhole Markup Language (KML), Scalable Vector Graphics (SVG), or a web-accessible resource like a JPEG or PNG picture.

Standardised raw data access is granted by the use of OGC WFS, WMS and WCS standards. These well-known standards provide access to data in various formats such as vectors (points, lines and polygons), raster images, and coverages (surface-like structures).

More about the described sensor related and data provision standards can be found on the OGC web site[1].

B. Location-aware Complex Event Processing

Apart from standardised data transmission and provision, a special focus in the Live Geography approach is the extension of Complex Event Processing (CEP) functionality by spatial parameters. In a geographic context, CEP can for instance serve for detecting threshold exceedances, for geo-

[1] http://www.opengeospatial.org

fencing implementations, for investigating spatial clusters, or for ensuring data quality.

Generally speaking, CEP is a technology that extracts knowledge from distributed systems and transforms it into contextual knowledge. Since the information content of this contextual knowledge is higher than in usual information, the business decisions that are derived from it can be more accurate. The CEP system itself can be linked into an Event Driven Architecture (EDA) environment or just be built on top of it. While the exact architecture depends on the application-specific needs and requirements, a general architecture including five steps can be applied to every CEP system. Fig. 3 shows this architecture in Job Description Language (JDL) [19].

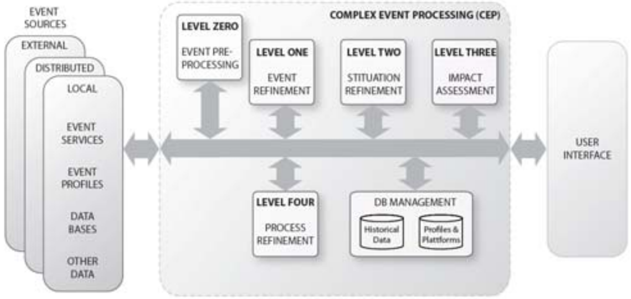


Figure 3. General CEP Architecture with Levels of Data Processing. [19]

The system is divided into several components (or levels) that represent processing steps. Since individual systems may have different requirements the implementation depth will vary from system to system. The following list notes the basic requirements for each level, according to [19]:

- *Level 0: Pre-processing* – Before the actual processing takes part the data is normalised, validated and eventually pre-filtered. Additionally, it may be important to apply feature extraction to get rid of data that is not needed. Although this part is important for the CEP workflow, it is not a particularity of CEP in general. Thus, it is marked as level 0.
- *Level 1: Event Refinement* – This component's task is to track and trace an event in the system. After an event has been tracked down, its characteristics (e.g. data, behaviour and relationships) are translated into event-attributes. The tracing part deals with state estimation that tries to predict upcoming events in the system.
- *Level 2: Situation Refinement* – The heart of each CEP system is the step of situation refinement, where the actual analysis of simple and complex events takes place. This analysis includes mathematical algorithms, which comprise not only computing boundaries for certain values, but also matching them against patterns and historical data. The results are high level (contextual) interpretations which can be acquired in real-time.
- *Level 3: Impact Assessment* – After analysing and refining the situation, it is important to generate decisions that may have consequences. Impact assessment deals with the simulation of outcomes. It therefore deals with various scenarios and simulates them by accounting for cost factors and resources. Results are weighted decision reports that include priorities and proposals for corresponding scenarios.
- *Level 4: Process Refinement* – Finally, the last step covers the interaction between the CEP system and business processes. It provides a feedback loop to control and refine business processes. While this could include integration and automatic controlling of processes, it may also just be the creation of alerting messages or business reports.

Originally, CEP has been developed and traditionally been implemented in the financial and economic sectors to predict market developments and exchange rate trends. In these areas, CEP patterns emerge from relationships between the factors *time*, *cause* (dependency between events) and *aggregation* (significance of an event's activity towards other events). In a location-aware CEP, these aspects get extended by additional parameters indicating *location* information of the event. Spatial parameters are combined on par with other relational indicators.

As geo-referenced data is therefore subject for further processing, one important aspect is to control its data quality. Quality criteria comprise lineage, logical consistency, completeness or temporal quality. From a practical viewpoint, this means that the location may be used to define quality indicators, which can be integrated into CEP pattern rules.

C. *Implementation*

The implementation of the Live Geography approach comprises tailor-made sensing devices, a real-time data integration mechanism, a use case specific interpolation model, an automatic server-based analysis component, and a complex event processing and alerting component. All these stand-alone modules are chained together using open standards as described below.

For the **measurement device**, we designed a special sensing pod for pervasive GIS applications using ubiquitous embedded sensing technologies. The system has been conceived in such a modular way that the base platform can be used for a variety of sensor web applications such as environmental monitoring, biometric parameter surveillance, critical infrastructure protection or energy network observation by simply changing the interfaced sensors.

The sensor pod itself consists of a standard embedded device, a Gumstix Verdex XM4 platform including an ARM7-based 400MHz processor with 64MB RAM and 16MB flash memory. It runs a customised version of the „Open Embedded" Linux distribution (kernel version 2.6.21) with an overall footprint of <8MB. Additionally to this basic operating system, we attached a GPS module (U-BLOX NEO 4S and LEA-4P) for positioning and several different sensors (e.g. LM92 temperature, NONIN 8000SM oxygen saturation and pulse, or SSM1 radiation sensors).

The software infrastructure comprises an embedded secure web server (Nostromo nhttpd), an SQLite database and several daemons, which convert sensor readings before they are served to the web. The database serves for short-term storage of historic measurements to allow for different error detection procedures and plausibility checks, as well as for non-sophisticated trend analysis.

For standardised data retrieval, we have created a service implementing the OGC Sensor Observation Service (SOS), SensorML and Observations and Measurements (O&M) standards in an application-specific way. Like this, measurement data are served via HTTP over Universal Mobile Telecommunications System (UMTS) in the standardised XML-based O&M format over the SOS service interface. SensorML serves for describing the whole sensor platform as well as to control the sensor via the OGC Sensor Planning Service (SPS), for instance to dynamically adjust measurement cycles.

The overall service stack on the embedded sensing device is illustrated in Fig. 4.

For the **real-time data integration component**, we have developed a data store extension to the open-source product *GeoServer*[2] 1.6. This plug-in enables the direct integration of OGC SOS responses (in O&M format) into GeoServer and their conversion to OGC-conformal service messages on-the-fly. The main advantage of this approach is that sensor data are made available through a variety of established geo-standardised interfaces (OGC WFS, WMS, WCS), which offer a variety of output formats such as OGC Geographic Markup Language (GML), OGC Keyhole Markup Language (KML), geoRSS, geoJSON, Scalable Vector Graphics (SVG), JPEG or PDF.

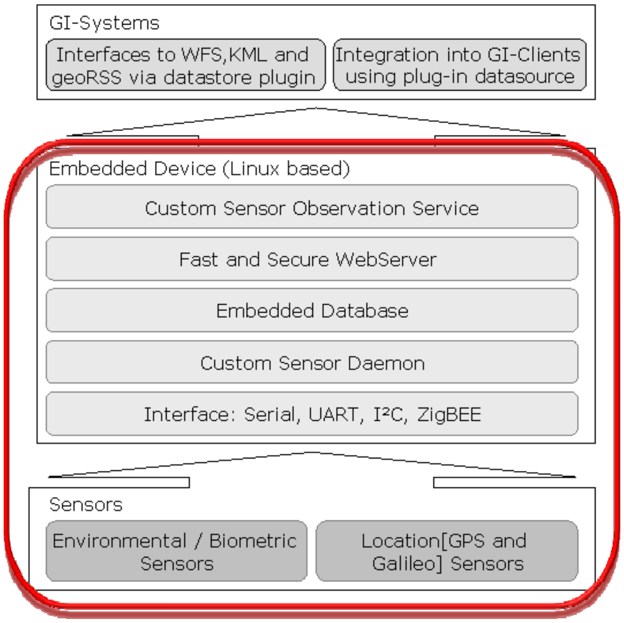

GI-Systems
Interfaces to WFS,KML and geoRSS via datastore plugin
Integration into GI-Clients using plug-in datasource
Embedded Device (Linux based)
Custom Sensor Observation Service
Fast and Secure WebServer
Embedded Database
Custom Sensor Daemon
Interface: Serial, UART, I²C, ZigBEE
Sensors
Environmental / Biometric Sensors
Location[GPS and Galileo] Sensors

Figure 4. Software Infrastructure on the Embedded Sensing Device.

During the transformation procedure from O&M input to WFS output, certain input parameters (coordinate reference system, unit conversions, data structures etc.) are interpreted. In practice, this means that the O&M XML structure is converted into well-established standardised data formats as mentioned above.

The innovation in comparison to previous data integration approaches is that the conversion of the data structure from SOS responses to various WFS, WMS and WCS output formats is performed on-the-fly. Conventional methods typically use the laborious interim step of storing data in a temporary database. This approach has two distinct disadvantages. At first, it adds another component to the overall workflow, which likely causes severe performance losings; secondly, it creates a single central point of failure making the system vulnerable in case of technical malfunctions.

On the contrary, our approach allows for the establishment of a distributed sensor service architecture, and thus enables data provision of real-time measurements for heterogeneous application domains and requirement profiles. The direct conversion of data structures allows for the easy integration of sensor data into GIS applications and therefore enables fast and ubiquitous data visualisation and analysis.

In its current implementation, **geographical analysis** is performed by ESRI's ArcGIS software suite since reliable open processing services are not yet available. We created a *Live Sensor Extension* that allows for the direct integration of standardised sensor data into ArcGIS. The *Live View* component enables ad-hoc GIS processing and visualisation in a navigable map using ESRI's Dynamic Display technology. Fig. 5 shows an interpolated temperature surface in ArcScene using the Inverse Distance Weighting (IDW) algorithm. In addition to ArcGIS, we have also used a number of open/free clients such as Open Layers, uDig, Google Maps, Google Earth or Microsoft Virtual Earth to visualise sensor information in 2D and 3D.

A second processing module implementing the Live Geography framework is an ArcGIS Tracking Analyst based spatio-temporal analysis component. For our study, we used CO_2 data captured by the CitySense [8] network in Cambridge, MA US.

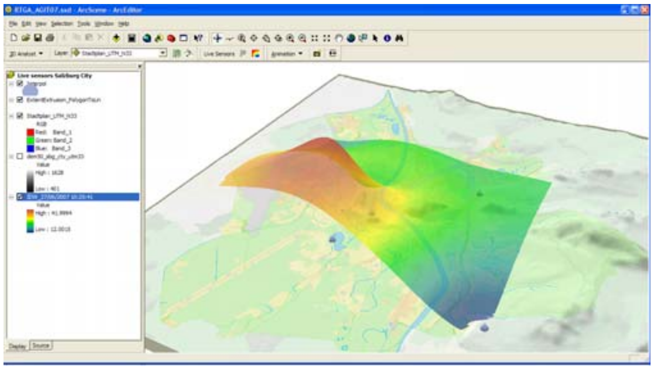


Figure 5. 3D Interpolation Using the Inverse Distance Weighting (IDW) Algorithm.

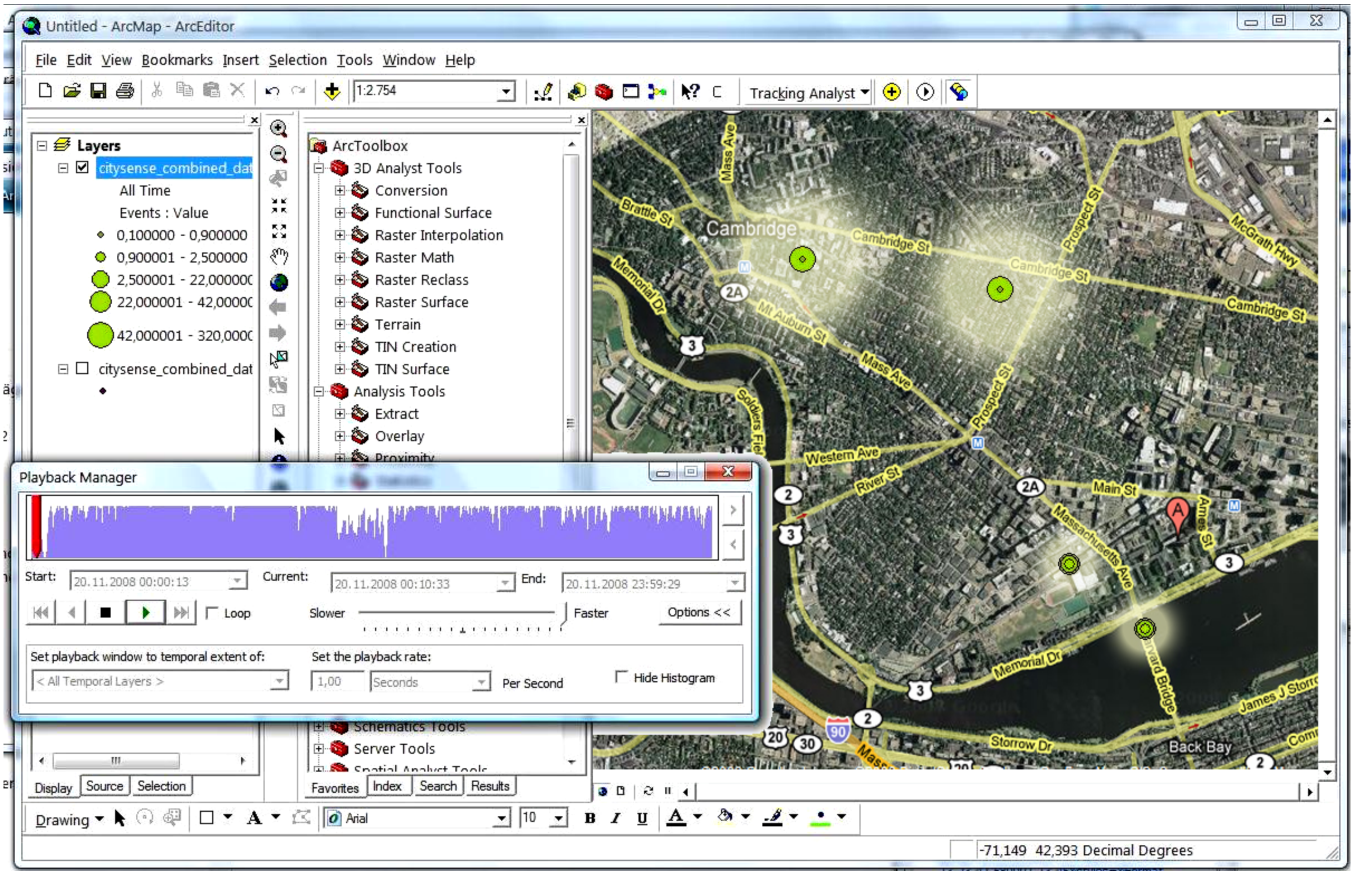
[2] http://www.geoserver.org

Fig. 6 shows the interface, which illustrates a time series of measurement data over a period of time. The lower left part of the figure shows the temporal gradient of the measurement values. Running the time series then changes symbologies in the map on the right side accordingly in a dynamic manner. Preliminary findings show that CO_2 is characterised by very high temporal and spatial fluctuations, which are induced by a variety of factors including temperature variability, time during the day, traffic emergence or „plant respiration”. In further analysis, this would allow for instance correlating temporal measurement data fluctuation to traffic density, weather conditions or day-time related differences in a very flexible way. Together with the Public Health Department of the City of Cambridge, we are currently carrying out more detailed investigations on these aspects.

A particularly innovative part of the implementation is the web-based GI processing component. We established two data analysis models for Inverse Distance Weighting (IDW) and Kriging operations. Together with the web interface source code itself, we then integrated these models into a single toolbox, which can be published as a web service on ArcGIS server. This allows for data analysis by just selecting base data and the according processing method in a two-click procedure. Two distinct advantages of this mechanism versus current desktop GIS solutions are that geographic analysis can be done without profound expert knowledge, and that processing requirements are shifted away from desktop computers to the server-side. These benefits will likely induce a paradigm shift in GI data processing in the next years and foster a broader spectrum of application areas for spatial analysis operations.

Furthermore, we implemented a **CEP and alerting mechanism** based on XMPP (Extensible Messaging and Presence Protocol), conformant to the OGC Sensor Alert Service (SAS) specification.

In our case, CEP is used for detecting patterns in measurement data and for creating complex events accordingly. These can be related to time (temporal validity), space (e.g. geo-fencing with geographic „intersect“, „overlap“, or „join“ operations), or measurement parameters (e.g. threshold exceedances). Event recognition and processing happens in two different stages of the workflow. Firstly, at sensor level CEP is used to detect errors in measurement values by applying different statistical operations such as standard deviations, spatial and temporal averaging, or outlier detection. Secondly, after the data harmonisation process CEP serves for spatio-temporal pattern recognition, anomaly detection, and alert generation in case of threshold transgression.



1:2.754

citysense_combined_dat
All Time
Events : Value
0,100000 - 0,900000
0,900001 - 2,500000
2,500001 - 22,00000C
22,000001 - 42,00000C
42,000001 - 320,000C
citysense_combined_dat

Start: 20.11.2008 00:00:13 Current: 20.11.2008 00:10:33 End: 20.11.2008 23:59:29

Set playback window to temporal extent of: < All Temporal Layers >

Set the playback rate: 1,00 Seconds Per Second

Arial 10

-71,149 42,393 Decimal Degrees

Figure 6. Time Series Analysis in Tracking Analyst.

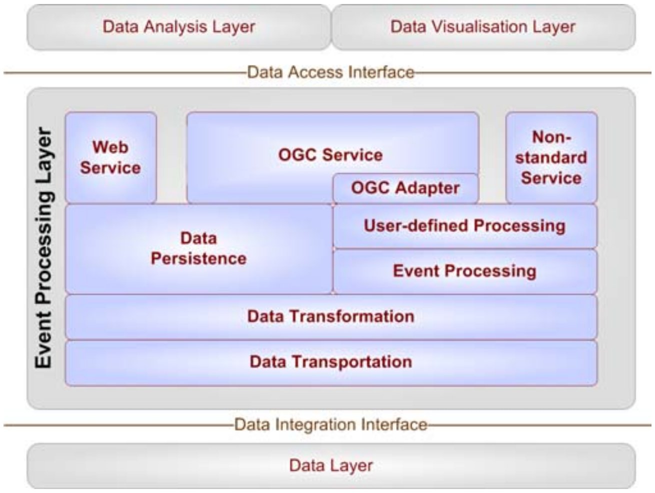


Figure 7. Internal Service Stack of the CEP Component.

In the actual implementation, we used the Esper CEP engine in its version 3.0.0 because of its open availability, Event Query Language (EQL) based event description, and its simple integration by just including a single Java library. For sending events created by the CEP engine, we realised a push-based OGC SAS compliant alerting service. SAS is an asynchronous service connecting a sensor in a network to an observation client. In order to receive alerts, a client subscribed to the SAS. If the defined rules apply, a pre-defined alert is sent to the client via XMPP. It shall be stated that the whole communication between the embedded XMPP server (jabberd2) and the client is XML-based for simplifying M2M messaging.

Fig. 7 shows the internal sub-parts of the CEP-based event processing component, which is built up in a modular structure. Generally speaking, the event processing component connects the data layer (i.e. sensor measurements), and the data analysis and data visualisation components. In other words, it prepares raw data in order to be process-able in the analysis and the visualisation layers. The data transportation component is responsible for connecting the data integration layer to a wide variety of data sources, which can comprise sensor data, real-time RSS feeds, ftp services, web services, databases etc. Hence, it serves as an entry point into the system. Its main responsibility is to receive various kinds of data structures and pass them on as is to the data transportation layer adding metadata do the actual payload, such as the data source or its format.

The event processing component handles the objects coming from the transformation module according to specified user query statements. This module basically handles multiple streams and identifies and selects meaningful events. Currently, the Esper event processing engine is used to detect events and push data to the user processing component. We extended the Esper engine by spatial parameters as described above. Following the event processing step, a further user-defined processing method is applied. This component implements a set of filtering or selection rules, which are specified by the user according to a very specific need.

The data persistence component receives a set of data from the data transformation module or from the processing modules (i.e. a „filtered“ dataset). From these data, it creates a physical data structure, which can either be temporary (for rapidly changing real-time data) or permanent (for time-insensitive data as created by stationary measurement devices). Consequently, as well live data sources as static ones can be handled by the data persistence component.

The non-standard service is one of three service interfaces, which connect the data integration layer to the data analysis layer. It provides data via a custom (i.e. non-standard) interface. This is necessary as existing standardised OGC services do not automatically support push mechanisms. It shall be mentioned that standardised data structures can also be served via the non-standard interface, just the service interface itself is not wholly standards-conformal. The service is also responsible for converting objects, which are created by the data transformation component, to a common, pre-defined output format. The output can either be GML, KML or geoRSS for spatial data, or RSS, JSON, SOAP bindings or a custom API for non-spatial data.

The OGC service is the second component, which connects the data integration layer to the data analysis layer. It offers a well-known and widely spread standardised interface in proved data structures such as GML, geoTIFF, KML or SVG. In essence, the main difference compared to the non-standard service is that this component provides OGC-standardised data structure over standardised service interface instead of providing a custom interface. The indicated OGC real-time adapter is basically a technological bridge to integrate live data into existing OGC services as existing implementations only support a variety of static data sources such as shape files, ASCII grids, different (geospatial) databases or cascading OGC services (WFS, WMS, WCS). Its implementation has been described earlier in this chapter.

The web service component is the third interface connecting the integration layer with the analysis layer. It is intended for handling non-geographic non-real-time data and for serving it via the http protocol. This component is basically a regular web service, meaning that it implements the request/response communication model, but no pushing mechanism as the non-standard service component does.

The next release of the implementation will enable a wide range of collection and reporting possibilities for integration into existing decision support systems in a variety of application areas, yielding a more complete and accurate real-time view of the city. Furthermore, the integration of Event Stream Processing (ESP) mechanisms will allow for management of pre-defined domain violations and for tracing and analysing spatio-temporal patterns in continuous stream data.

VI. Effects on the Urban Context

From a socio-political viewpoint, Live Geography is primarily targeted at the information needs of local and regional governments. It enables them to respond to the environmental and social challenges of the city and to learn more about the impacts of urban policies and practices. Moreover, it also supports professionals, such as urban and transportation planners, in building or refining their models of urban dynamics. In fact, it can change the work of practitioners that was previously about predicting and accommodating, and which is now becoming more observing and improving. Indeed, this new ability to render all kinds of „machine readable“ environments not only provide new views on the city and its environment, but also supply urban and transportation engineers and planners with indicators to evaluate their interventions. For instance, Dan Hill and Duncan Wilson foresee the ability to tune buildings and cities through pre and post occupancy evaluations. They speculate that the future of environmental information will be part of the fabric of buildings [20]. However, this integration opens all sorts of issues regarding sampling, density, standardisation, quality control, power control, access to data, and update frequency.

A complete picture might be hard to achieve with incomplete environmental data patched together by data mining, filtering and visualisation algorithms. Environmental monitoring in the urban context is limited to classic technical issues related to data resolution and heterogeneity. Even mobile sensors do not yet provide high-density sampling coverage over a wide area, limiting research to sense what is technically possible to sense with economical and social constraints. One set of solutions rely on the calibration of mathematical models with only a few sensors nodes and complementing data sources to create a set of spatial indicators. Another, approach aims at revealing instead of hiding the incompleteness of the data. Visualising the uncertainty of spatial data is a recurrent theme in cartography and information visualisation [21]. These visualisation techniques present data in such a manner that users are made aware of the degree of uncertainty so as to allow for more informed analyses and decisions. It is a strategy to promote the user appropriation of the information with an awareness of its limitations [22]. Without these strategies to handle the fluctuating quality of the data, their partial coverage could impact people's perception of the environment, by providing a quasi-objective but inaccurate angle of the content, and potentially „negatively“ influencing their behaviour.

Another way to improve the coverage of environment data is to alter the current model whereby civic government would act as sole data-gatherer and decision-maker by empowering everyday citizen to monitor the environment with sensor-enabled mobile devices. Recently providers of geographic and urban data have learned the value of people-centric sensing to improve their services and from the activities of their customers. For instance the body of knowledge on a city's road conditions and real-time road traffic network information thrive on the crowd-sourcing of geo-data the owners of TomTom system and mobile phone operators customers generate. Similarly, the users of Google MyMaps have contributed, without their awareness, to the production the massive database necessary for the development of the location-based version of the application. However, this people-centric approach to gather data raise legitimate privacy, data integrity and accuracy concerns. These issues can be handled with a mix of policy definition, local processing, verification and privacy preserving data mining techniques [23]. These technical solutions necessitate a richer discussion beyond the academic domain on these observing technologies' social implications.

Similar crowd-sourcing strategies have been considered for environmental monitoring with individuals acting as sensor nodes and coming together with other people in order to form sensor networks. Several research projects explore a wide range of novel physical sensors attached to mobile devices empowering everyday non-experts to collect and share air quality data measured with sensor-enabled mobile devices. For instance, Ergo [24] is a simple SMS system that allows anyone with a mobile phone to quickly and easily explore, query, and learn about local air quality on-the-go with their mobile phone. With these tools, citizens augment their role, becoming agents of change by uncovering, visualising, and sharing real-time air quality measurements from their own everyday urban life. This „citizen science“ approach [25] creates value information for researchers of data generated by people going on their daily life, often based on explicit and participatory sensing actions. By turning mobile phones [26], watches [27] or bikes [28] into sensing devices, the researchers hope that public understandings of science and environmental issues will be improved and can have access to larger and more detailed data sets. This access to environmental data of the city also becomes a tool to raise the citizen awareness of the state of the environment.

These data gathering possibilities also imply that we are at the end of the ephemeral; in some ways we will be able to replay the city. In contrast we are also ahead of conflicts to reveal or hide unwanted evidences, when new data can be used to the detriment of some stakeholder and policy makers. Indeed, the capacity to collect and disseminate reconfigure sensor data influence political networks, focussing on environmental data as products or objects that can be used for future political action. Therefore, openness, quality, trust and confidence in the data will also be subject of debate (e.g. bias to have people record their observations, who gets to report data and who not). This people-centric view of measuring, sharing, and discussing our environment might increase agencies' and decision makers' understanding of a community's claims, potentially increasing public trust in the information provided by a Live Geography approach.

This raises a real challenge: how can we encourage, promote and motivate environmentally sustainable behaviours on the basis of Live Geography information? Recent work [29] shows how real-time information about cities and their patterns of use, visualised in new ways, and made available locally on-demand in ways that people can act upon, may make an important contribution to sustainability. However, the communication of data collected from pervasive sensor networks may not trigger sufficient motivation for people to change their habits towards a more environmentally sustainable lifestyle. Indeed, this objective of improving the environmental sustainability of a city calls for behaviour modification. It can be induced by intervening in moments of local decision-making and by providing people with new rewards and new motivations for desirable behaviours [30]. These kinds of strategies have been common, for instance, in health and fitness applications. However, when we think about persuasion in the real of environment sustainability, we might want to persuade people of the ways, in which their interests are aligned with those of others [31]. Therefore, this process of alignment and mobilisation, by which one can start to find one's own interests as being congruent with those of others will be critical in the success of these strategies based on Live Geography.

All of the above examples consider the use of sensed data to satisfy public or private requirements of some sort. However, terms like „air quality“ are effectively only a surrogate for the health effects of pollutants on people or structures, and there is much disagreement on the public information regulations and whether they are effective. One potential alternative is the direct sensing of structural, or even human, impacts. In other words, people need to be sensitised to their location and their environment, and this information will have to be presented within different contexts, such as public safety, energy consumption or sustainability in order to open new possibilities for exploring the city. In sum, feedback of „sensed“ data to the public – either directly or after processing – can potentially change people's perception of the city itself. For example, weather or pollution prediction on a very local scale in space and time is reasonably feasible. Continuous availability of this information could lead to a change in people's individual behaviour by giving real-time short-term decision support. The Live Geography approach can play a significant role in achieving this seminal vision.

VII. Conclusion

Ubiquitous and continuous environmental monitoring is a multi-dimensional challenge, and this is particularly true in the urban context. In this paper we have shown which issues have to be considered for environmental monitoring systems in the city, and have outlined how the *Live Geography* approach can meet these requirements.

It stands for the combination of live measurement data with historic data sources in an open standards based framework using server-side processing mechanisms. Its basic aim is twofold: first, automating GIS analysis processes that currently require a considerable amount of manual input, using new server-based processing tools by providing a wholly standardised workflow; second to replace monolithic measurement systems with an open standards-based infrastructure.

The implementation of the approach comprises the following components: firstly, an embedded measurement device providing sensor data via the standardised OGC Sensor Observation Service (SOS) interface; secondly, a special sensor fusion mechanism to harmonise these data, in our case a GeoServer custom data store. The component provides live measurements in well-established standardised formats (KML, GML, geoRSS etc.). This enables simple integration into specialised GIS analysis software.

A crucial implementation component is the Complex Event Processing (CEP) module, which serves for detecting different patterns – i.e. „events“ – in measurement data (threshold exceedance, geo-fencing, etc.), and for quality assurance. The module creates alerts, which are sent via the OGC Sensor Alert Service (SAS) interface. According to the alert, different actions can be taken. Either a message (SMS, email etc.) is sent to subscribers of the alert, or geo-analysis operations is triggered, e.g. a flow model calculation in case of threshold exceedance of local rivers.

Performance of these server-based analysis tasks is satisfactory for near real-time decision support. For instance, a standard Kriging interpolation of 14 sensors takes about 6.4 seconds with an output resolution of 50m using an area of interest of 7x5km, using ArcGIS Server version 9.3. It shall be noted that the largest part (approximately 80%) of the delay is due to sensor response times.

To prove the system's portability, we deployed the same underlying framework in different application areas: environmental monitoring (air temperature variation assessment), public health (air quality and associated health effects) and patient surveillance (monitoring of biometric parameters). The next practical realisation will be an urban „quality of life“ monitoring system in order to gain a city-wide picture of spatial and temporal variations of environmental parameters such as particulate matter, pollen concentrations or CO pollution. Their integration with static GIS data (census, traffic emergence, living vs. working places etc.) will maximise significance for local and regional governments as well as for citizens by applying complex GIS algorithms such as kriging or co-kriging to reveal unseen causal correlations between spatial parameters. Other scheduled implementations comprise radiation monitoring and water quality assessment infrastructures. To achieve enhanced usability on the end user side, we are just developing a mobile GIS application, which allows the user to interact with the system while offering a broad range of geographic analysis tools.

Since interoperable open systems in general are not trivial to implement, this motivation has to be initiated through legal directives like INSPIRE, GMES, SEIS and GEOSS. As all of our *Live Geography* implementations are operated in close cooperation with local or regional governments and thematic actors, we think that the Live Geography approach will raise awareness of ubiquitous sensing systems and perhaps trigger profound rethinking process in collaboration and cooperation efforts between different authorities in the city.

Concluding, it shall be stated that the trend towards extensive availability of measurement data requires a paradigm shift in the global GIS market and its applications for urban environmental monitoring. This applies especially to open data accessibility and intensified collaboration efforts. To achieve far-reaching adoption, the establishment of ubiquitous sensing infrastructures will require a long process of sensitising individuals to their spatial and social contexts, and to how to connect local environmental questions to well-known urban issues such as public safety, energy efficiency, or social interaction. In effect, creating a meaningful context around densely available sensor data and distributing specific information layers makes the environment more understandable to the city management, to the citizens and to researchers by „making the abstract real", i.e. by revealing hidden connections in real-time.

ACKNOWLEDGMENT

Our approach requires expertise in a wide variety of research areas such as sensor networks, data integration, GIS data and analysis, visualisation techniques etc. We would like to thank all contributing groups at the Research Studio iSPACE and at MIT for their valuable inputs and contributions in different stages of the project.

REFERENCES

[1] Resch, B., Mittlboeck, M., Girardin, F., Britter, R. and Ratti, C. (2009) Real-time Geo-awareness - Sensor Data Integration for Environmental Monitoring in the City. IN: Proceedings of the IARIA International Conference on Advanced Geographic Information Systems & Web Services – GEOWS2009, 1-7 February 2009, Cancun, Mexico, pp. 92-97.

[2] Paulsen, H. and Riegger, U. (2006). SensorGIS – Geodaten in Echtzeit. In: GIS-Business 8/2006: pp. 17-19, Cologne.

[3] Resch, B., Calabrese, F., Ratti, C. and Biderman, A. (2008) An Approach Towards a Real-time Data Exchange Platform System Architecture. In: Proceedings of the 6th Annual IEEE International Conference on Pervasive Computing and Communications, Hong Kong, 17-21 March 2008.

[4] University of Oklahoma (2009) OKCnet. http://okc.mesonet.org, March 2009. (12 May 2009)

[5] Center for Coastal and Land-Margin Research (2009) CORIE. http://www.ccalmr.ogi.edu/CORIE, June 2009 (14 July 2009)

[6] Kansal, A., Nath, S., Liu, J. and Zhao, F. (2007) SenseWeb: An Infrastructure for Shared Sensing. IEEE Multimedia, 14(4), October-December 2007, pp. 8-13.

[7] Paulsen, H. (2008) PermaSensorGIS – Real-time Permafrost Data. Geoconnexion International Magazine, 02/2008, pp. 36-38.

[8] Murty, R., Mainland, G., Rose, I., Chowdhury, A., Gosain, A., Bers, J. and Welsh, M. (2008) CitySense: A Vision for an Urban-Scale Wireless Networking Testbed. Proceedings of the 2008 IEEE International Conference on Technologies for Homeland Security, Waltham, MA, May 2008.

[9] National Oceanic and Atmospheric Administration (2008) nowCOAST: GIS Mapping Portal to Real-Time Environmental Observations and NOAA Forecasts. http://nowcoast.noaa.gov, September 2008. (15 December 2008)

[10] Sarjakoski, T., Sester, M., Illert, A., Rystedt, B., Nissen, F. and Ruotsalainen, R. (2004) Geospatial Info-mobility Service by Real-time Data-integration and Generalisation. http://gimodig.fgi.fi, 8 November 2004. (22 May 2009)

[11] Rittman, M. (2008) An Introduction to Real-Time Data Integration. http://www.oracle.com/technology/pub/articles/rittman-odi.html, 2008. (22 May 2009)

[12] Sybase Inc. (2008) Real-Time Events Data Integration Software. http://www.sybase.com/products/dataintegration/realtimeevents, 2008. (22 December 2008)

[13] Rahm, E., Thor, A. and Aumueller D. (2007) Dynamic Fusion of Web Data. XSym 2007, Vienna, Austria, pp.14-16.

[14] Riva, O. and Borcea, C. (2007) The Urbanet Revolution: Sensor Power to the People!. IEEE Pervasive Computing, 6(2), pp. 41-49, April-June 2007.

[15] Resch, B., Schmidt, D. und Blaschke, T. (2007) Enabling Geographic Situational Awareness in Emergency Management. In: Proceedings of the 2nd Geospatial Integration for Public Safety Conference, New Orleans, Louisiana, US, 15-17 April 2007.

[16] Botts, M., Percivall, G., Reed, C. and Davidson, J. (Eds.) (2007a) OGC® Sensor Web Enablement: Overview and High Level Architecture. http://www.opengeospatial.org, OpenGIS White Paper OGC 07-165, Version 3, 28 December 2007. (17 August 2009)

[17] Resch, B., Mittlboeck, M., Lipson, S., Welsh, M., Bers, J., Britter, R. and Ratti, C. (2009) Urban Sensing Revisited – Common Scents: Towards Standardised Geo-sensor Networks for Public Health Monitoring in the City. In: Proceedings of the 11th International Conference on Computers in Urban Planning and Urban Management – CUPUM2009, Hong Kong, 16-18 June 2009.

[18] Schut, Peter (ed.) (2007) Web Processing Service. http://www.opengeospatial.org, OpenGIS Standard, Version 1.0.0, OGC 05-007r7, 8 June 2007. (19 June 2009)

[19] Bass, T. (2007) What is Complex Event Processing?. http://www.thecepblog.com/what-is-complex-event-processing, 2007. (11 July 2009)

[20] Hill, D. and Wilson, D. (2008) The New Well-tempered Environment: Tuning Buildings and Cities. Pervasive Persuasive Technology and Environmental Sustainability, 2008.

[21] MacEachren, A. M., Robinson, A., Hopper, S. Gardner, S., Murray, R., Gahegan, M. and Hetzler, E. (2005) Visualizing Geospatial Information Uncertainty: What We Know and What We Need to Know. Cartography and Geographic Information Science, 32(3), pp. 139–160, July 2005.

[22] Chalmers, M. and Galani, A. (2004) Seamful Interweaving: Heterogeneity in the Theory and Design of Interactive Systems. DIS'04: Proceedings of the 2004 Conference on Designing Interactive Systems, ACM Press, New York, NY, USA, 2004, pp. 243–252.

[23] Abdelzaher, T., Anokwa, Y., Boda, P., Burke, J., Estrin, D., Guibas, L., Kansal, A., Madden, S. and Reich, J. (2007) Mobiscopes for Human Spaces. IEEE Pervasive Computing, 6(2), pp. 20–29, 2007.

[24] Paulos, E. (2008) Urban Atmospheres - Proactive Archeology of Our Urban Landscapes and Emerging Technology. http://www.urban-atmospheres.net, 2008. (18 June 2009)

[25] Paulos, E., Honicky, R. and Hooker, B. (2008) Citizen Science: Enabling Participatory Urbanism. In: Handbook of Research on Urban Informatics: The Practice and Promise of the Real-Time City. 506 pp., ISBN 9781605661520, IGI Global, Hershey PA, USA, 2009.

[26] Nokia Research (2008) SensorPlanet. http://www.sensorplanet.org, 12 March 2008. (18 August 2009)

[27] La Montre Verte (2009) La Montre Verte City Pulse. http://www.lamontreverte.org, July 2009. (10 August 2009)

[28] Campbell, A.T., Eisenman, S.B., Lane, N.D., Miluzzo, E. and Peterson, R.A. (2006) People-centric Urban Sensing. WICON'06: Proceedings of the 2nd Annual International Workshop on Wireless Internet, ACM, New York, NY, USA, 2006.

[29] Calabrese, F., Kloeckl, K. and Ratti, C. (2008) WikiCity: Real-time Location-sensitive Tools for the City. In: Handbook of Research on Urban Informatics: The Practice and Promise of the Real-Time City. 506 pp., ISBN 9781605661520, IGI Global, Hershey PA, USA, 2008.

[30] Fogg, B.J. (2003) Persuasive Technology: Using Computers to Change What We Think and Do. 312 pp., ISBN 978-1558606432, Morgan Kaufmann, Amsterdam, The Neatherlands, 2003.

[31] Dourish, P. (2008) Points of Persuasion: Strategic Essentialism and Environmental Sustainability. Proceedings of the Sixth International Conference on Pervasive Computing, Sydney, Australia, 19-22 May 2008.

Rewards and Risks in P2P Content Delivery

Raimund K. Ege
Northern Illinois University
Dept. of Computer Science
DeKalb, IL
ege@niu.edu

Li Yang
Dept. of Computer Science
University of Tennessee
Chattanooga, TN
Li.Yang@utc.edu

Richard Whittaker
School of Comp. & Info Science
Florida International University
Miami, FL
rwhitt01@cs.fiu.edu

Abstract—*The ever increasing speed of access to the Internet has enabled sharing of data on an unprecedented scale. Data of all forms and shapes is becoming easily accessible: large multi-media files are being routinely downloaded onto a plethora of end user devices. Peer-to-peer content delivery approaches enable massive scale in the amount of data volume that can be efficiently delivered. The openness of delivery demands adaptive and robust management of intellectual property rights. In this paper we describe a framework and its implementation to address the central issues in content delivery: a scalable peer-to-peer-based content delivery model, paired with an access control model that balances trust in end users with a risk analysis to the data provider. Our framework enables data providers to extract the maximum amount of return, i.e. value, from making their original content available. Our implementation architecture provides a protocol to leverage the greatest amount of reward from the intellectual property that is released to the Internet.*

Keywords-broadband file sharing; peer-to-peer content delivery; intellectual property rights for multi media

I. INTRODUCTION

Making multimedia content available online has become a Killer-Application for the Internet. Services such as iTunes, YouTube, Joost and Hulu are popularizing delivery of audios and video content to anybody with a broadband internet connection. Additionally, virtual communities are emerging (such as FaceBook, MySpace and Twitter) where users communicate directly with one another to exchange information or execute transactions in a peer-to-peer fashion. These services are currently struggling with the challenges of securing large-scale distribution. The dynamism of peer-to-peer communities means that principals who offer services will meet requests from unrelated or unknown peers. Peers need to collaborate and obtain services within environment that is unfamiliar or even hostile. Therefore, peers have to manage the risks involved in the collaboration when prior experience and knowledge about each other are incomplete. One way to address this uncertainty is to develop and establish trust among peers. Trust can be built by either a trusted third party [2] or by community-based feedback from past experiences [3] in a self-regulating system. Trust leads naturally to a decentralized approach to security management that can scale up in size, but must be balanced with a measure of risk that is the flip side of trust.

Conventional approaches rely on well-defined access control models [4, 5] that qualify peers and determine authorization based on predefined permissions. In such a complex and collaborative world, a peer can protect and benefit itself only if it can respond to new peers and enforce access control by assigning proper privileges to new peers. The nature of digital content requires access models that go beyond checking authorization upon initial access: authorization variables quickly change in a dynamic context. The Usage Control Model (UCON) [6] is an example of a framework to handle continuity of access decisions and mutability of subject and object attributes. Authorization decisions are made before an access, and repeatedly checked during the access. On-going access may be revoked if security policies are violated. The more dynamic the situation is the more likely access will be denied, therefore denying the data provider any benefit.

The general goal of our work is to address both the trust in peers which are allowed to participate in the content delivery process, and quantifying the risk and reward garnered from releasing data in to the network. We investigate the design of a novel approach to access control. If successful, this approach will offer significant benefits to emerging peer-to-peer applications. It will also benefit collaboration over the existing Internet when the identities and intentions of parties are uncertain. We integrate trust evaluation for usage control with an analysis of risk/reward. Underlying our framework is a formal computational model of trust and access control that will provide a formal basis to interface authentication with authorization.

Our paper is organized as follows; the next section will explain our approach to peer-to-peer content delivery. Section III will elaborate on how the data source and its peers can quantify gain from participating in the content delivery. Section IV explains our risk/reward model that enables a data source to initially decide on whether to share the content and keep some leverage after its release. Section V gives an overview of our implementation framework, and Section VI details the prototype implementation of our framework that employs the fairly new Stream Control Transmission Protocol (SCTP) which improves over the current stand-bearers Transmission Control Protocol (TCP) and User Datagram Protocol (UDP) for multi-stream session-oriented delivery of large multi-media files over fast networks. The paper concludes with our perspective on how modern content delivery approaches will usher in a new generation of Internet applications. An earlier version of this paper was presented at the Fourth International Conference on Systems (ICONS 2009), Cancun, Mexico, March 2009 [1].

II. PEER to PEER CONTENT DELIVERY

Peer-to-peer (P2P) delivery of multimedia aims to deliver multi-media content from a source to a large number of clients. For our framework, we assume that the content comes into existence at a source. A simple example of creating such multimedia might be a video clip taken with a camera and a microphone, or more likely video captured via a cell phone camera, and then transferred to the source. Likewise the client consumes the content, e.g. by displaying it on a computing device monitor, which again might be a cell phone screen watching a YouTube video. We further assume that there is just one original source, but that there are many clients that want to receive the data. The clients value their viewing experience, and our goal is to reward the source for making the video available.

In a P2P delivery approach, each client participates in the further delivery of the content. Each client makes part or all of the original content available to further clients. The clients become peers in a peer-to-peer delivery model. Such an approach is specifically geared towards being able to scale effortlessly to support millions of clients without prior notice, i.e. be able to handle a “mob-like” behavior of the clients.

The exact details of delivery may depend on the nature of the source data: for example, video data is made available at a preset quality using a variable-rate video encoder. The source data stream is divided into fixed length sequential frames: each frame is identified by its frame number. Clients request frames in sequence, receive the frame and reassemble the video stream which is then displayed using a suitable video decoder and display utility. The video stream is encoded in such a fashion that missing frames don’t prevent a resulting video to be shown, but rather a video of lesser bit-rate encoding, i.e. quality, will result [7]. We explicitly allow the video stream to be quite malleable, i.e. the quality of delivery need not be constant and there is no harm if extra frames find their way into the stream. It is actually a key element of our approach that the stream can be enriched as part of the delivery process.

In our approach, multi-media sources are advertised and made available via a central tracking service: at first, this tracker only knows the network location of the server. Clients that want to access the source do so via the tracker: they contact the tracker, which will respond with the location of the source. The tracker will also remember (or track) the clients as potential new sources of the data. Subsequent client requests to the tracker are answered with all known locations of sources: the original and the known client. Clients that receive locations of sources from the tracker issue frame requests immediately to all sources. As the sources delivery frames to the clients, the client stores them. The client then assumes a server role and also answers requests for frames that they have received already, which will enable a cascading effect, which establishes a P2P network where each client is a peer. Every client constantly monitors the rate of response it gets from the sources and adjusts its connections to the sources from which the highest throughput rate can be achieved.

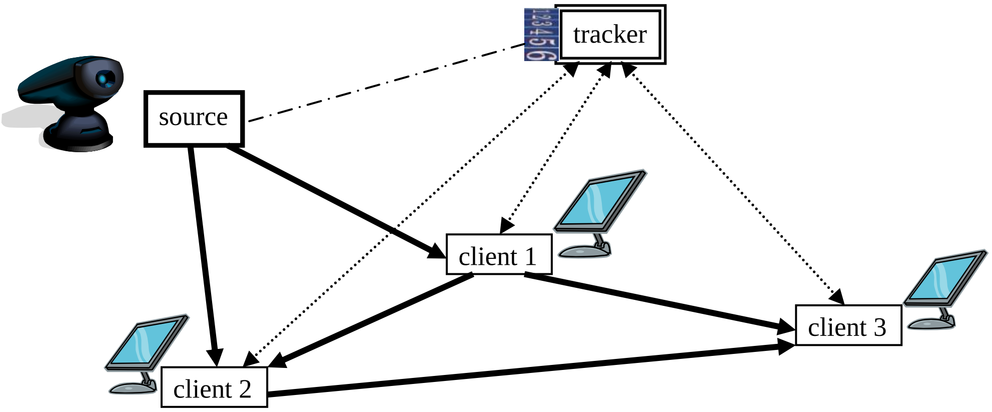


Figure 1. Content delivery network

Figure 1 shows an example snapshot of a content delivery network with one source, one tracker and three clients. The source is where the video data is produced, encoded and made available. The tracker knows the network location of the source. Tracker and source maintain a secure connection. Clients connect to the tracker first and then maintain sessions for the duration of the download: all 3 clients maintain an active connection to the tracker. The tracker informs the client which source to download from: Client 1 is fed directly from the source; client 2 joined somewhat later and is now being served from the source and client 1; client 3 joined last and is being served from client 1 and client 2. In this example, two of the clients are also serving as intermediaries on the delivery path from original source to ultimate client.

III. UNITS of RISK and REWARD

We assume that the data made available at the source has value. Releasing the data to the Internet carries potential for reaping some of the value, but also carries the risk that the data will be consumed without rewarding the original source. There is also a cost associated with releasing the data, i.e. storage and transmission cost. For example, consider a typical "viral" video found on YouTube.com: the video is uploaded onto YouTube.com for free, stored and transmitted by YouTube.com and viewed by a large audience. The only entity that is getting rewarded is YouTube.com, which will accompany the video presentation with paid advertising. The person that took the video and transferred it to YouTube.com has no reward: the only benefit that the original source of the video gets is notoriety.

In order to provide a model or framework to asses risk and reward, we need to quantize aspects of the information interchange between the original source, the transmitting medium and the final consumer of the data. In a traditional fee for service model the reward *"R"* to the source is the fee *"F"* paid by the consumer minus the cost *"D"* of delivery:

$$R = F - D$$

The cost of delivery *"D"* consist of the storage cost at the server, and the cost of feeding it into the Internet. In the case of YouTube, considerable cost is incurred for providing the necessary server network and their bandwidth to the Internet. YouTube recovers that cost by adding paid advertising on the source web page as well as adding paid advertising onto the video stream. YouTube's business model recognizes that these paid advertisings represent significant added value. As soon as we recognize that the value gained is not an insignificant amount, the focus of the formula shifts from providing value to the original data source to the reward that can be gained by the transmitter. If we quantify the advertising reward as *"A"* the formula now becomes:

$$R = F - (D - A)$$

Even in this simplest form, we recognize that *"A"* has the potential to outweigh *"D"* and therefore reduce the need for *"F"*. As YouTube recognizes, the reward lies in *"A"*, which is paid advertising that accompanies the video.

In our prior work we focused on mediation frameworks that capture the mutative nature of data delivery in the Internet [8, 9]. As data travels from a source to a client on lengthy path, each node in the path may act as mediator. A mediator transforms data

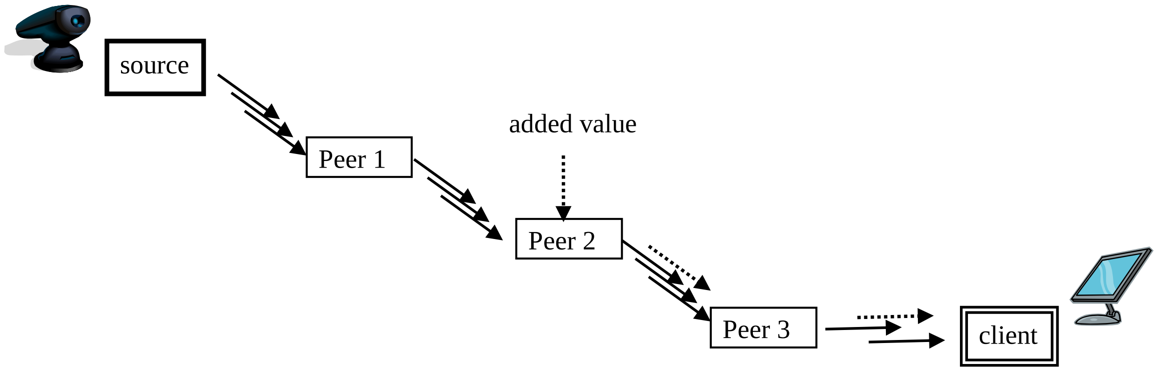


Figure 2. Varying peer behavior

from an input perspective to an output perspective. In the simplest scenario, the data that is fed into the delivery network by the source and is received by the ultimate client unchanged: i.e. each mediator just passes its input data along as output data. However, that is not the necessary scenario anymore: the great variety of client devices already necessitate that the data is transformed to enhance the client's viewing experience. We apply this mediation approach to each peer on the path from source to client. Each peer may serve as a mediator that may transform the content stream in some fashion. Our implementation employs the stream control transmission protocol (SCTP) which allows multi-media to be delivered in multiple concurrent streams. All a peer needs to do is add an additional stream for a video overlay message to the content as it passes through.

Figure 2 shows a sample path from the content source to its consumer. The multi-media source is fed as a multi stream into the content delivery path. Each peer on the path receives a number of streams and will do its best to deliver the streams to the next peer. "Peer 1" is an example of a peer that copies its input faithfully to its output. "Peer 2" shows a peer that adds an additional overlay stream to its output. Peer 3 is an example of a peer that filters out a stream to make its output more suitable for a specific target device.

The formula for reward can now be extended into the P2P content delivery domain, where a large number of peers serve as the transmission/storage medium. Assuming "n" number of peers that participate and potentially add value the formula is now:

$$R \;=\; F - \sum_{i=1}^{n}(D_i - A_i)$$

D_i and A_i are now the delivery cost and value incurred at each peer that participates in the P2P content delivery. The reward available to the data originator is potentially very large given the number of peers that be involved. A second dimension is opened up when we consider that the data will be consumed by many clients, so that the ultimate reward formula is the sum of all rewards gained from each client "c":

$$R_c \;=\; F_c - \sum_{i=1}^{n}(D_i - A_i)$$

Whether or not the data originator will gain any reward depends on whether the client pays fee "F" and whether the peers are willing to share their gain from the added value. In a scenario where clients and peers are authenticated and the release of the data is predicated by a contractual agreement, the source will reap the complete benefit.

In our model we quantify the certainty of whether the client and peers will remit their gain to the source with a value of trust *"T"*: T represents the trust in the client that consume that data, T represents the trust in each peer that participates in the content delivery:

$$R = \sum_{c}\left(F_c - \sum_{i=1}^{n}(D_i - A_i)\right) * T_c$$

The formula captures the ultimate truth that no reward will be materialized when there is no arrangement for trust.

IV. TRUST MODEL

It comes down to the question whether to accept a new peer into the content delivery network. For every request from a peer a measure of trust, i.e. a trust value, must be computed. The trust is evaluated based on both actual observations and recommendations from referees. Observations are based on previous interactions with the peer. Recommendations may include signed trust-assertions from other principals, or a list of referees that can be contacted for recommendations. The trust value, calculated from observations and recommendations, is a value within the [0, 1] interval evaluated for each peer that requests to be part of the content delivery.

The trust is assumed to follow a beta distribution, and is represented by the two parameters of the beta distribution. The beta distribution, a conjugate prior, is chosen because of its reproducibility property under the Bayesian framework. When a conjugate prior is multiplied with the likelihood function, it gives a posterior probability having the same functional form as the prior, thus allowing the posterior to be used as a prior in further computations. For a given requester, we define a sequence of variables T_1, T_2 ,…, T_k to characterize the trust at sampling time k.

For instance, at k^{th} sampling time, N_k observations were collected about the peer. Let G_k be the number of normal requests or behaviors. If there is no history of malicious behavior by the peer associated with a request (i.e., neither malware nor spyware were observed), the request is deemed as normal behavior.

Now suppose a prior probability density function (*pdf*) of trust T_{k-1}, denoted by f_{k-1}(t), is known about the peer. Then the posterior *pdf* of trust for this peer (given N_k = n and G_k = g) can be obtained from Bayes theorem [10, 11] as follows:

$$f_k(t) = \frac{f_k(g!\,t,n) f_{k-1}(t)}{\int_0^1 f(\,g!\,t,n)\, f_{k-1}(t)\; dt}$$

where $f_k(g!\,t,n)$ is called the likelihood function and has the form of a binomial distribution:

$$f_k(g!\,t,n) = \binom{n}{g} t^g (1-t)^{n-g}$$

The prior *pdf* f_{k-1}(t) summarizes what is known about the distribution of T_{k-1}. Under the assumption that prior *pdf* f_{k-1}(t) follows a beta distribution, it can be shown that the posterior *pdf* also follows a beta distribution.

In particular, if $f_{k-1}(t) \sim beta(\alpha_{k-1}, \beta_{k-1})$, we have $f_k(t) \sim beta(\alpha_{k-1} + g_k, \beta_{k-1} + n_k - g_k)$ given that N_k = n_k and G_k = g_k. Therefore, $f_k(t)$ is characterized by the parameters α_k and β_k defined recursively as follows: $\alpha_k = \alpha_{k-1} + g_k$ and $\beta_k = \beta_{k-1} + n_k - g_k$. Initially, there is no knowledge about the peer: we assume that trust values follow a uniform distribution of the interval [0,1], i.e. f_o (t) $\sim U[0,1] = beta(1,1)$ which indicates our ignorance about the new peer's behavior at time 0. Time 0 is when the peer first becomes known to the content delivery network.

At time k, trust value T_c for a given peer c is now:

$$T_c = \frac{\alpha_k}{\alpha_k + \beta_k}$$

There are two alternative ways to update trust values. One is to update trust values based on all known observations and recommendations. The other ways is to update trust values based on recent information only. The advantage of the latter one is two folds: reduce the computation complexity and detect a change in the peer's behaviors early. For instance, if a peer has been misbehaving for a short time period, then recent observations together with actual reports are more reflective of the behavior change than would be if trust was based on all available observations.

Meanwhile, recommendations from referees bring in new information T_{rq} on the peer's behaviors. We combine the new data T_{rq} with our own observation T_{oq} on the condition that the referee is highly trusted or the recommendation passes the deviation test. The deviation test is to decide whether a recommendation is trustworthy or not. Recommendation R is learned from past interactions the referee had with the requestor. Trustworthiness of a recommendation also follows a beta distribution. f_k(t) is adjusted by recommendations: $T_{oq} := T_{oq} + \mu T_{rq}$ where T_{oq} is trust we have in the peer, T_{rq} is trust that the referee has in the peer, and µ is the trust in the referee's recommendations.

In summary, when it comes down to the question whether to accept a given peer into the content delivery network, we now have a tool to assess the potential gain balanced by the risk posed by the new peer:

$$R = \sum_c \left(F_c - \sum_{i=1}^{n} (D_i - A_i) \right) * T_c$$

Our model correlates the reward gained from accepting the new peer with the risk posed by the new peer and to enable an informed decision.

V. IMPLEMENTATION FRAMEWORK

Peer-to-peer networks are open by definition. While being open, ready to give access, our BitTorrent-style of delivery uses a tracker approach. The tracker keeps information that is global to the data exchange and can be the place to gather and disseminate control information. Each BT-style distribution of original content requires at least one tracker. Additional trackers can easily be established. The first and subsequent trackers need to carry trust with the original data source. If there is more than one tracker, we use a public key infrastructure approach[12] to authenticate and certify each tracker. The number of trackers needed is small and will pose little overhead to our model.

The tracker is the location where the decision on which peers may participate in the content delivery is made. While it may seem that the original source should be the decision maker, for the purpose of our model the original source is assumed to have delegated this authority to the tracker(s).

Our framework therefore features 3 types of participants:

1. tracker, where all information on the current status of the content delivery network is maintained and all access decisions are made.
2. client, where the consumption of the data occurs.
3. source, where the data is available for further dissemination. The original source is the first source. Clients that have downloaded and consumed the data will immediately become new sources.

What we called "peer" in our discussion so far, starts out by requesting access from the tracker, then becomes a client and ultimately a new source for that data that is being tracked and access-controlled by the tracker(s).

A. Client

The key to a smooth scaling of this ad-hoc p2p network is the algorithm used by the client to request frames from a source (either the original source or another client). A client consists of three processes:

1) a process to communicate with the tracker. The client initiates the negotiation with the tracker to enable the tracker's decision on whether the peer is admitted into the content delivery network. Upon success, the tracker informs the client which sources the client should use, and the client will update the tracker on its success in downloading the source data;
2) a process to request data from the given sources. Since the original data may be very large and exist in multiple fragments. For example, video data is typically made available as a series of frames. Fragments or frames may be requested from multiple sources. The bittorrent protocol uses algorithms to determine which sources are most likely to yield the best throughput; and
3) a process to receive frames/fragments from sources and to assemble them into usable data.

All three processes share the following data:

- a list of recommended sources to download from. This list is originally received from the tracker, but can be modified by the client based on download success;
- a list of backup servers to download from, received from the tracker, but the client can move servers from this list to the list of recommended servers based on the client's download success;
- current bandwidth utilization at the client.

The client continuously requests fragment/frame sequences until the end of the transmission is reached. It requests a fragment/frame sequences from each available source. The "receive" process runs in a continuous loop that accepts frames from servers. While frames within a sequence will arrive in the correct order the receiving process still needs to order the frame sequences number. The process also records which server delivered the frames.

The most vital process in the client is its communication with the tracker. At first, the communication focuses on qualifying the client for participation the content delivery network. During the ongoing download, the communication is meant to validate the client's continued credentials. Since trust in the peer is calculated on a continuing basis, the client needs to be in constant communication with the tracker. No lapses in the continuous communication are allowed.

B. Source

A peer that is admitted into the content delivery network will operate initially as a client. As soon as sufficient data has been downloaded, the tracker will determine whether the client can also serve as source for further downloads. The decision depends on the amount of data the client holds, and which portions of the original data are already collected in the client. Once the tracker determines that the client can serve as source, further negotiation is necessary to assess which added values the new source will contribute to the calculation of reward:

$$R = \sum_{c} \left(F_c - \sum_{i=1}^{n} (D_i - A_i) \right) * T_c$$

The tracker needs to know the values of *"D"* and *"A"* that this new source will incur. *"D"* is the cost of transmission and storage that the new source will have to pay, whereas *"A"* is the added value that the new source might be able to realize by being part of the delivery network. Based on the outcome of the negotiation with the tracker, the new source will then serve as a new mediator for the original data. The higher the trust *"T"* is that the tracker places onto the new source, the higher to overall reward will become for the original data owner. However, even a smaller amount of trust will realize an additional gain for the original source. In addition, ongoing monitoring of downloading peers must be maintained.

C. Tracker

The core of the content delivery model is the tracker. While the tracker might initially be a single unit, it can easily be duplicated, as long as a strong trust relationship is maintained between the trackers, and continuous exchange of peer information is maintained.

The tracker is first enabled by the original source of the data content. The tracker knows the location of the original/first source. The tracker starts by initializing its database of peers. The initial state of peer database can be augmented by historical data and/or a distributed-hash-table style if control data dissemination.

Peers that wish to participate in the content delivery must first locate the tracker. Public directories are the usual places where trackers are listed. Search engines exist for the sole purpose of publicizing content that is available for download.

A peer will start by establishing a connection to a tracker. The tracker will consider the request from a new peer and gather the necessary data on the trust in the new peer. The tracker will seek information to establish the peer's trust value:

$$T_c = \frac{\alpha_k}{\alpha_k + \beta_k}$$

If the peer is new and not yet listed in the tracker(s) database, then a new entry is created. The tracker will also determine the new peer's contribution to the reward formula. The peer will contribute a value *"D"* and *"A"*, to reflect the additional cost and added value. The tracker is the location where the determination is made whether the gain possible from admitting the new peer outweighs the risk of releasing the data content to an untrusted peer. Only if the overall reward formula shows a potential gain, then is the new peer accepted. Initially, the peer is admitted as client. As the peer accumulates downloadable volume, the tracker may elevate the status to create a new source that is allowed to provide new added content.

VI. PROTOTYPE IMPLEMENTATION

Our current prototype is implemented using the Java programming language. Since we are using the newly standardized SCTP protocol, we require the use of the OpenJDK version 7 [13] which is currently undergoing beta evaluation. To enable truly large numbers of truly large frames in our multimedia content delivery network, we keep all elements of the implementation in the 64bit space. Unfortunately 64bit implementations of SCTP are not yet standard within the Microsoft Windows family of operating system, so we are currently limited to running our prototype elements on Linux 64bit operating systems that provide direct kernel support for the new protocol via the lksctp [14] library.

SCTP [15] is a Transport Layer protocol, serving in a similar role as the popular TCP and UDP protocols. It provides some of the same service features of both, ensuring reliable, in-sequence transport of messages with congestion control.

We chose SCTP because of its ability to delivery multimedia in multiple streams. Once a client has established a SCTP association with a server, packages can be exchanged with high speed and low latency. Each association can support multiple streams, where the packages that are sent within one stream are guaranteed to arrive in sequence. Each source can divide the original video stream into set of streams meant to be displayed in an overlay fashion. Streams can be arranged in a way that the more streams are fully received by a client, the better the viewing quality will be. When sending a packet over a SCTP channel we need to provide an instance of the MessageInfo class, which specifies which stream the packet belongs to. The first stream is used to deliver a basic low quality version of the video stream. The second and consecutive streams will carry frames that are overlaid onto the primary stream for the purpose of increasing

```
01 SocketAddress socketAddress = new InetSocketAddress(port);
02 channel =  SctpMultiChannel.open().bind(socketAddress);
03 MessageInfo info;
04 while ((info = channel.receive(bb, null, null)) != null) {
05   // determine requestor
06   Association association = info.association();
07   // determine which frame range
08   bb.flip();
09   int fromFrame = bb.getInt();
10   int toFrame = bb.getInt();
11   // send frames to requestor
12   for (int i=fromFrame; i<= toFrame; i++) {
13     bb.clear();
14     bb.putInt(i);
15     bb.put(framePool.getFrame(i));
16     bb.flip();
17     channel.send(bb,
              MessageInfo.createOutgoing(association, null,0));
18   }
19 }
```

Figure 3. SctpMultiChannel maintains one-to-many association

the quality. In our framework we also use the additional streams to carry content that is “added value”, such as advertising messages or identifying logos. The ultimate client that displays the content to a user will combine all streams into one viewing experience.

The second feature of SCTP we use is its new class “SctpMultiChannel” which can establish a one-to-many association for a single server to multiple clients. The SctpMultiChannel is able to recognize which client is sending a request and enables that the response is sent to that exact same client. This is much more efficient than a traditional “server socket” which for each incoming request spawns a subprocess with its own socket to serve the client. Figure 3 shows the Java source code where an incoming request is received. Each packet that is received on the channel carries a MessageInfo object which contains information on the actual client that is the actual other end point of this association. The Java code on line 06 retrieves the “association” identity from the incoming message “info” instance. The association is then used to send the response via the same SctpMultiChannel instance but only to the actual client that had requested the frames. The code on line 17 shows that a new outgoing message info instance is created for the same “association” that carried the incoming request. The message info instance is then used to send the response packet to the client. The code to receive SctpMultiChannel packets is logically similar to any UPD or TCP style of socket receive programming. Figure 4 shows a sample.

```
SocketAddress socketAddress =
          new InetSocketAddress(peer.address, peer.port);
SctpChannel channel = SctpChannel.open(socketAddress, 1, 1);
// send requested frame range to peer
ByteBuffer byteBuffer = ByteBuffer.allocate(128);
byteBuffer.putInt(fromFrame);
byteBuffer.putInt(toFrame);
byteBuffer.flip();
channel.send(byteBuffer, MessageInfo.createOutgoing(null, 0));
// here is where we read response
byteBuffer = ByteBuffer.allocate(64000);
while ((channel.receive(byteBuffer, null, null)) != null) {
   byteBuffer.flip();
   int frame = byteBuffer.getInt();
   System.out.print("Message received: " + frame);
   …
```

Figure 4. Packets from SctpMultiChannel being received by client

The three major components of the framework are implemented as “SourceMain”, “TrackerMain” and “ClientMain”, which are composed from classes that implement the core behavior of maintaining communication sessions, accepting requests for frames and delivering them, and requesting and receiving frames. The major classes are FrameRequestor and FrameServer. The original source starts out as the sole instance of FrameServer. The first client starts out as the sole instance of FrameRequestor. As the client accumulates frames it then also instantiates a FrameServer that is able to receive requests from other clients. A client that contains both a FrameRequestor and FrameServer instance becomes a true peer in the P2P content delivery framework.

In summary, tracker, source and client together contribute to build a highly efficient delivery network.

VII. CONCLUSION

In this paper we have described a model and framework for a new generation of content delivery networks. Our framework is designed enable content originators to assess the potential reward from distributing the content to the Internet. The reward is quantified as the value added at each peer in the content delivery network and gauged relative to the actual cost incurred in data delivery but also correlated to the risk that such open delivery poses. We described an implementation architecture that follows a bittorrent-style of P2P network, where a tracker disseminates information on which sources are available to download from. This information is constantly updated and communicated to new clients. New clients join the content delivery network and become new sources for new clients to download from. Such P2P content delivery has great potential to enable large scale delivery of multimedia content.

Consider the scenario we described earlier in the paper: a typical “viral” video found on YouTube.com: the video is uploaded onto YouTube.com for free, stored and transmitted by YouTube.com and viewed by a large audience. The only entity that is getting a reward is YouTube.com, which will accompany the video presentation with paid advertising. The only benefit that the original source of the video gets is notoriety.

Using our model, the original data owner can select other venues to make the video available via a peer-to-peer approach. The selection on who will participate can be based on how much each peer contributes in terms of reward but also risk. Peers will have an interest in being part of the delivery network, much like YouTube.com has recognized its value. Peers might even add their own value to the delivery and share the proceeds with the original source.

Whereas in the YouTube.com approach the reward is only reaped by one, and the original source has shouldered all the risk, i.e. lost all reward from the content, our model will enable a more equitable mechanism for sharing the cost and reward. Our model might just enable a new and truly openness of content delivery via the Internet.

REFERENCES

[1] Raimund K. Ege, Li Yang, Richard Whittaker. Extracting Value from P2P Content Delivery. Proceedings of the Fourth International Conference on Systems (ICONS 2009), pages 102-108 Cancun, Mexico, March 2009.

[2] Y. Atif. Building trust in E-commerce. IEEE Internet Computing, 6(1):18–24, 2002.

[3] P. Resnick, K. Kuwabara, R. Zeckhauser, and E. Friedman. Reputation systems. Communications of the ACM, 43(12):45–48, 2000.

[4] E. Bertino, B. Catania, E. Ferrari, and P. Perlasca. A logical framework for reasoning about access control models. In SACMAT ’01: Proceedings of the sixth ACM symposium on Access control models and technologies, pages 41–52, New York, NY, USA, 2001.

[5] S. Jajodia, P. Samarati, M. L. Sapino, and V. S. Subrahmanian. Flexible support for multiple access control policies. ACM Transaction Database System, 26(2):214–260, 2001.

[6] J. Park and R. Sandhu. The UCON usage control model. ACM Transaction Information System Security, 7(1):128–174, 2004.

[7] C. Wu, Baochun Li. R-Stream: Resilient peer-to-peer streaming with rateless codes. In Proceedings of the 13th ACM International Conference on Multimedia, pages 307-310, Singapore, 2005.

[8] R. Whittaker, G. Argote-Garcia, P. Clarke, R. Ege, Optimizing Secure Collaboration Transactions for Modern Information Systems, Proceedings of the Third International Conference on Systems (ICONS 2008), pages 62-68, Cancun, Mexico, 2008.

[9] R. K. Ege, L. Yang, Q. Kharma, and X. Ni. Three-layered mediator architecture based on dht. Proceedings of the 7th International Symposium on Parallel Architectures, Algorithms, and Networks (I-SPAN 2004), Hong Kong, SAR, China. IEEE Computer Society, pages 317–318, 2004.

[10] L. Yang, R. Ege, Integrating Trust Management into Usage Control in P2P Multimedia Delivery, Proceedings of Twentieth International Conference on Software Engineering and Knowledge Engineering (SEKE'08), pages 411-416, Redwood City, CA, 2008.

[11] A. Papoulis. Probability, Random Variables, and Stochastic Processes. McGraw-Hill, New York, 1991.

[12] Gutmann, P., 1999. The Design of a Cryptographic Security Architecture, *Proceedings of the 8th USENIX Security Symposium*, pages 153-168, Washington, D.C., 1999.

[13] java.net – The Source for Java Technology Collaboration, The JDK 7 Project, http://jdk7.dev.java.net. [accessed September 22, 2009]

[14] The Linux Kernel Stream Control Transmission Protocol (lksctp), a SourceForge project to provide SCTP for the Linux kernel, http://lksctp.sourceforge.net/. [accessed September 22, 2009]

[15] R. Stewart (ed.), Stream Control Transmission Protocol, Request for Comments: 4960, IETF Network Working Group, September 2007, http://tools.ietf.org/html/rfc4960. [accessed September 22, 2009]

Improving the Quality of Control of Periodic Tasks Scheduled by FP with an Asynchronous Approach

P. Meumeu Yomsi, L. George, Y. Sorel, D. de Rauglaudre
AOSTE Project-team
INRIA Paris-Rocquencourt
Le Chesnay, France
$\{$*patrick.meumeu, laurent.george, yves.sorel, daniel.de_rauglaudre*$\}$*@inria.fr*

Abstract

The aim of this paper is to address the problem of correctly dimensioning real-time embedded systems scheduled with Fixed Priority (FP) scheduling. It is well known that computers which control systems are greatly affected by delays and jitter occurring in the control loop. In the literature, a deadline reduction approach has been considered as one solution to reducing the jitter affecting a task, thereby obtaining better loop stability in the control loop. Here, in order to improve the sensitivity of the deadlines, we propose another solution for reducing the worst case response time of the tasks, hence reducing the jitter, when all the tasks are scheduled with the Deadline Monotonic Algorithm. This is performed for a specific asynchronous scenario for harmonic periodic tasks. We compare the results to those for the synchronous scenario in terms of minimum deadline reduction factor preserving the schedulability of tasks set in both cases.

Keywords: Real-time systems, Fixed-priority scheduling algorithms, Sensitivity analysis, Robust control.

1. Introduction

In this paper we consider the problem of correctly dimensioning real-time embedded systems ([1], [2], [3], [4]). The correct dimensioning of a real-time system strongly depends on the determination of the tasks' Worst-Case Execution Times (WCETs). Based on the WCETs, Feasibility Conditions (FCs) ([5], [6], [7]) can be established to ensure that the timeliness constraints of all the tasks are always met when tasks are scheduled by a fixed or a dynamic priority driven scheduling algorithm. We consider an application composed of a periodic task set $\Gamma_n = \{\tau_1, \cdots, \tau_n\}$ of n periodic tasks, scheduled with Fixed Priority (FP) preemptive scheduling. The classical definition of a periodic task τ_i, is:

- C_i: the Worst Case Execution Time (WCET) of τ_i.
- T_i: the period of τ_i.
- D_i: the relative deadline of τ_i (a task requested at time t must be terminated by its absolute deadline $t + D_i$), where $D_i \leq T_i$.

A recent research area called sensitivity analysis aims at providing interesting information on the validity of feasibility conditions by considering possible deviations of task WCETs ([2]), task periods ([2]), or task deadlines ([3]). This makes it possible, for example, to find a feasible task set, if the current one is not feasible, by modifying the task parameters or determining the impact of a change in architecture on the feasibility of a task set. A task set is declared feasible if for any task in the synchronous scenario, its worst case response time is less than or equal to its deadline. We are interested in the sensitivity of deadlines. Computer controlling systems are very much affected by delays and jitter occurring in the control loop. A deadline reduction has been considered by ([8]) as one solution to reducing the jitter affecting a task and therefore obtaining better loop stability in the control loop. The jitter of a task depends on the minimum and on the worst case response times. Reducing the deadline of a task can be a way to reduce the worst case response time of a task and thus can reduce the jitter of the task. However, this deadline reduction should be performed in such a way that it does not cause any task to fail at run-time. This supposes a scheduling driven by deadlines.

This paper proposes a solution to reduce as much as possible the worst case response time of each task when tasks are scheduled with fixed priorities, according to Deadline Monotonic Algorithm, by using a specific asynchronous first release times scenario. We show the benefits of our asynchronous scenario by comparing the minimum deadline reduction factor applied preserving the schedulability of the tasks in the synchronous and in the our asynchronous scenario.

With Deadline Monotonic Algorithm, tasks are scheduled according to their relative deadlines. The smaller the relative deadline, the higher the priority. Starting from a schedulable task set, we want to characterize the minimum deadline reduction factor $0 < \alpha \leq 1$ such that any task $\tau_i, i = 1, \ldots, n$ having a deadline $D_i = \alpha \times T_i$ is schedulable. α is such that any smaller reduction factor would lead to a non schedulable task set. We compare the value of α obtained in the worst case synchronous scenario (all the tasks are first released at the same time) to that obtained with a particular asynchronous scenario that we propose, and which has some interesting properties. We show that the minimum reduction factor obtained in our asynchronous scenario is always less than or equal to the minimum reduction factor obtained in the synchronous scenario.

Reducing the deadline of a task makes it possible to reduce the jitter resulting from the execution of a task. In this paper we show that the maximum deadline reduction is minimized for the synchronous scenario where all the tasks are first released at the same time.

We then propose a particular asynchronous first release times scenario that allows us to obtain better feasibility conditions and a better deadline reduction factor than the one obtained with the synchronous scenario, thus reducing the jitter of the tasks for a better control.

The feasibility problem of asynchronous task sets is known to be more complex than for synchronous task sets. We introduce a new formalism to compute the worst case response time of a task for asynchronous task sets. We apply this approach to the case where the periods of the tasks are harmonic. We then show that in this case, the worst case response time is always obtained for the second instance of a task, which represents a significant reduction in complexity.

The rest of the paper is organized as follows. In section 2, we give a state of the art regarding sensitivity analysis of deadlines considering dynamic and fixed priority driven schedulings. We then focus on asynchronous task sets and recall existing feasibility conditions. In section 3, we introduce the concepts and notations and establish important properties for the particular asynchronous scenario that we have chosen. We consider harmonic periods. We show that, using this particular scenario, the worst case response time of every task is obtained for its second instance[1]. In section 4, we introduce a new scheduling representation which is more compact than the classical linear representation / Gantt Chart for a schedule. In section 5, we introduce the concept of Mesoid which is used to compute the worst case response time of an asynchronous task set. In section 6, we give an algorithm for the computation of the worst case response time of any task in our asynchronous scenario, then we show how to compute the minimum deadline reduction factor. An example is given in order to compare the deadline reduction factor obtained with our asynchronous scenario to that in the synchronous scenario. We provide experimental results in section 7 based on extensive simulations comparing the deadline reduction factor for several load configurations in both the synchronous case and in our asynchronous scenario. Finally, we conclude in section 8.

2. State of the art

Sensitivity analysis for deadlines has been considered for Earliest Deadline First (EDF) scheduling algorithm by ([3]) showing how to compute the minimum feasible deadlines such that the deadline of any task τ_i equals αD_i, where α is reduction factor $0 < \alpha \leq 1$. In ([8]), the space of feasible deadlines (D-space), a space of n dimensions has been considered. Any task set having deadlines in the D-space is considered to be schedulable. To the knowledge of the authors, no work has been done on the sensitivity of deadlines for fixed priority scheduling algorithms.

Few results have been proposed to deal with the deadline assignment problem. As far as the authors are aware, no results are available for Fixed Priority (FP) scheduling. Baruah & al., in [9] propose modifying the deadlines of a task set to minimize the output, seen as a secondary criteria. In Cervin & al. ([10]), the deadlines are modified to guarantee close-loop stability of a real-time control system. Marinca & al. ([11]) focus on the deadline assignment problem in the distributed case for multimedia flows. The deadline assignment problem is formalized in terms of a linear programming problem. The scheduling considered on every node is non-preemptive EDF or FIFO, with a jitter cancellation applied on every node. A performance evaluation of several deadline assignment schemes is proposed.

A recent paper proposed by Balvastre & al. ([3]) proposes an optimal deadline assignment for periodic tasks scheduled with preemptive EDF in the case of deadlines less than or equal to periods. The goal is to find the minimum deadline reduction factor preserving all the deadlines of the tasks.

They first focus on the case of a single task deadline reduction and show how to compute D_i^{min}, the minimum deadline of task τ_i such that any deadline smaller than D_i^{min} for task τ_i will lead to a non-feasible task set.

They also show in [3] that when considering the reduction of a single task τ_i, D_i^{min} is the worst case response time of task τ_i for EDF scheduling. The maximum deadline reduction factor α_i for task τ_i is then: $\alpha_i = 1 - \frac{D_i^{min}}{D_i}$.

In the case of a deadline reduction applied to n tasks, the goal is to minimize all tasks' deadlines assuming the same reduction factor for all the tasks (with no preference regarding which task requires the greatest deadline reduction). Balbastre & al. in [3] show how to compute the maximum deadline reduction factor α applied to all the deadlines using an iterative algorithm. The principle is to compute the minimum slack $t - h(t)$ for any time $t \in [0, L)$ to determine the deadline reduction factor applied to all the tasks, where $h(t) = \sum_{i=1}^{n} max(0, 1 + \lfloor \frac{t - D_i}{T_i} \rfloor) C_i$ and L is the length of the first synchronous busy period, solution of the equation $t = \sum_{i=1}^{n} \left\lceil \frac{t}{T_i} \right\rceil C_i$.

[1] Throughout the paper all subscripts refer to tasks whereas all superscripts refer to instances.

```
$\tau = \{\tau_1, \ldots, \tau_n\}$ : task set;
L $\leftarrow$ compute-L($\tau$) : integer;  $\alpha \leftarrow 1$ : real
$slack = min_{t\in[0,L)}(t - h(t))$ : real;
While ($slack \neq 0$) do
    $\alpha = min_{i=1\ldots n}(1 - \frac{slack}{D_i})$;
    For ($i = 1$;$i < n$;$i++$) do
        $D_i = \alpha D_i$;
    end For
    $slack = min_{t\in[0,L)}(t - h(t))$;
done
Return $\alpha$;
```

Algorithm 1: Computation of α for EDF scheduling

For Fixed Priority (FP) scheduling, necessary and sufficient FCs have been proposed in the case of non-concrete tasks where the first release times of the tasks can be arbitrary. A classical approach is based on the computation of the tasks' worst-case response times ([12], [6]). The worst-case response time, defined as the worst case time between the request time of a task and its latest completion time, is obtained in the worst case synchronous scenario where all the tasks are first released at the same time, and is computed by successive iterations. This worst case response time provides a bound on the response time valid for any other task first release times. It can be shown that considering only non-concrete tasks can lead to a pessimistic dimensioning [13].

The complexity of this approach depends on the worst case response time computation complexity. In the case of deadlines less than or equal to periods for all tasks, the worst-case response time R_i of a task τ_i is obtained in the synchronous scenario for the first release of τ_i at time 0 and is the solution of the equation ([12]) $R_i = W_i(R_i)$, where $W_i(t) = C_i + \sum_{\tau_j\in hp(i)} \left\lceil \frac{t}{T_j} \right\rceil C_i$ and $hp(i)$ denotes the set of tasks with a priority higher than or equal to that of τ_i except τ_i itself. The value of R_i is computed by successive iterations and the number of iterations is bounded by $1+\sum_{\tau_j\in hp(i)} \left\lfloor \frac{D_i}{T_j} \right\rfloor$. A necessary and sufficient feasibility condition for a task set is: $\exists t \in S$, such that $W_i(t)/t \leq 1$, where $S = \cup_{\tau_j\in hp(i)}\{kT_j, k \in N\} \cap [0, D_i]$. For any task τ_i, the checking instants correspond to the arrival times of the tasks with a higher priority than τ_i within the interval $[0, D_i]$. This feasibility has been improved by ([14]), where the authors show how to reduce the time instants of S. For any task τ_i, they show how to significantly reduce the number of checking instants during the interval $[0, D_i]$ to at most 2^{i-1} times rather than $1+\sum_{\tau_j\in hp(i)} \left\lfloor \frac{D_i}{T_j} \right\rfloor$. When deadlines and periods are independent, ([6]) shows that the worst-case response times of a sporadic task τ_i are not necessarily obtained for the first activation request of τ_i at time 0. The number of activations to consider is $1 + \left\lfloor \frac{L_i}{T_i} \right\rfloor$, where L_i is the length of the worst-case level-τ_i busy period defined in ([15]) as the longest period of processor activity running tasks of priority higher than or equal to τ_i in the synchronous scenario. It can be shown that $L_i = \sum_{\tau_j\in hp(i)\cup\tau_i} \left\lceil \frac{L_i}{T_j} \right\rceil C_j$. From its definition, L_i is bounded by:

$$Min \left\{ \sum_{\tau_j\in hp(i)\cup\tau_i} \frac{C_j}{1 - \sum\limits_{\tau_j\in hp(i)\cup\tau_i} \frac{C_j}{T_j}}, \sum_{\tau_j\in hp(i)\cup\tau_i} \frac{C_j}{T_j} \cdot P \right\} ([7]).$$

where $P = LCM(T_1, \ldots, T_n)$ is the least common multiple (*LCM*) of the periods of all tasks and it leads to a pseudo-polynomial time complexity for the feasibility conditions.

This is an interesting approach as it provides a pseudo-polynomial time complexity but it may lead to a pessimistic dimensioning as the synchronous scenario might not be likely to occur.

In order to improve the schedulability of the systems, offset strategies on the first release times of the tasks have been considered. A system where offsets are imposed is called an asynchronous system. ([13]) shows significant feasibility improvements considering offsets. Simulations show that the number of feasible schedulable systems with offsets (while unfeasible in the synchronous case) increases with the number of tasks for a processor load of 0.8 and ranges from 40.5% to 97% for different offset assignment strategies. This percentage strongly decreases when the load is high (tends to 1).

With asynchronous tasks, ([16]) shows that for a given offset assignment, the schedulability of the tasks must be checked in the interval $[0, max_{i=1\ldots n}(O_i)+2P)$ where P is the least common multiple of the tasks and O_i is the offset of task τ_i, leading to an exponential time complexity. To provide less pessimistic FCs, it is furthermore mandatory to prove that the offsets will not result later in a synchronous scenario. This problem is referred to as the K-simultaneous congruence problem in the state of the art ([16]). This feasibility result has been significantly improved by ([17]) showing that the interval to check the feasibility of a periodic task set with offsets can be reduced to $[0, max_{i=1\ldots n}(O_i) + P)$.

Furthermore, ([16]) proves the non optimality of Deadline Monotonic scheduling algorithm for asynchronous systems when task deadlines are less than or equal to periods. An optimal priority assignment can be obtained in $O(n^2)$ using the Audsley procedure ([18]).

A particular case denoted *offset free systems* corresponds to the case where offsets can be chosen arbitrarily. An optimal offset assignment is given in ([19]). An offset assignment is optimal if it can find a schedulable offset whenever a feasible assignment exists. The complexity of the offset assignment algorithm is exponential and is in $O((max_{2\leq j\leq n}T_j)^{n-1})$. The offset of task τ_1 is set to 0. Different offset strategies / heuristics have been considered in the literature. Among them, we can cite the dissimilar offset assignment proposed by ([19]) that consists in shifting (computing a distance between the offsets) the offset of the tasks to be as far as possible from the synchronous scenario. The algorithm sorts the couple of tasks (τ_i, τ_j) by decreasing values of $gcd\{T_i, T_j\}$ such that the distance belongs to $[0, gcd\{T_i, T_j\})$. The dissimilar offset assignment significantly reduces the number of offsets to consider, leading to a complexity in $O(n^2.(log(max_{i\in[1,n]}T_i)+log(n^2)))$. Other offset assignment strategies have been considered by ([13]) using the Audsley procedure to determine the subset of tasks of τ that can be feasibly scheduled in the synchronous scenario (setting their offset to 0). The offsets are only computed for the subset of tasks that are unfeasible with the Audsley procedure in the synchronous case. The authors consider different criteria to assign the offsets, based on the criteria used to sort the couple of tasks (τ_i, τ_j). The complexity is the same as that of the dissimilar offset assignment.

In this paper we consider a particular asynchronous harmonic concrete task set where $\forall 2 \leq i \leq n, T_{i-1} \mid T_i$ (i.e. there exists $k \in \mathbb{Z}$ such that $T_i = kT_{i-1}$) with particular offsets. In the case of non-concrete harmonic tasks, when tasks are scheduled with Rate Monotonic Algorithm (the shorter the period, the higher the priority) and in the case where deadlines are equal to periods, a necessary and sufficient condition for the feasibility of such a system is given by $U = \sum_{i=1\ldots n} \frac{C_i}{T_i} \leq 1$ (see [20]). This potentially proves the benefits of considering harmonic tasks in order to get better feasibility conditions. This property does not hold when deadlines can be shorter than periods. In this case we show how to determine in $\mathcal{O}(n)$ the offset of the tasks to obtain a pseudo-polynomial time feasibility condition instead of an exponential one. In the case of asynchronous tasks, the worst case response time cannot be computed with a recursive equation as for the synchronous tasks. This is due to the fact that with offsets, there is not necessarily a continuous busy period from time 0 to the release time of a task. In this paper we investigate a new approach to compute the worst case response time of a task based on the Mesoid approach. This approach was first introduced by ([4]) in the context of real-time scheduling with preemption cost. This approach does not require a continuous busy period to compute the worst case response times of the tasks. We propose a particular offset assignment, such that the worst case response time of any task is obtained for its second request time, providing an exponential time improvement in the complexity of the FCs.

More recently, for control systems, [21] has proposed to include the control delay resulting from the response time of a task as a cost function for the controllers. They show how to solve the optimal period assignment problem analytically.

3. Properties of the asynchronous harmonic task set

3.1. Concepts and notations

We recall classical results in the uniprocessor context for real-time scheduling.

- Time is assumed to be discrete (task arrivals occur and task executions begin and terminate at clock ticks; the parameters used are expressed as multiples of the clock tick); in [22], it is shown that there is no loss of generality with respect to feasibility results by restricting the schedules to be discrete, once the task parameters are assumed to be integers (multiples of the clock tick) i.e. a discrete schedule exists, if and only if a continuous schedule exists.
- A task set is said to be valid with a given scheduling policy if and only if no task occurrence ever misses its absolute deadline with this scheduling policy.
- $U = \sum_{i=1}^{n} \frac{C_i}{T_i}$ is commonly called the processor utilization factor associated to the task set Γ_n, i.e., the fraction of processor time spent in the execution of the task set ([23]). If $U > 1$, then no scheduling algorithm can meet the tasks' deadlines.
- The synchronous scenario corresponds to the scenario where all the tasks are released at the same time (at time 0).

The model depicted in figure 1 is Liu & Layland's pioneering model [23] for systems executed on a single processor.

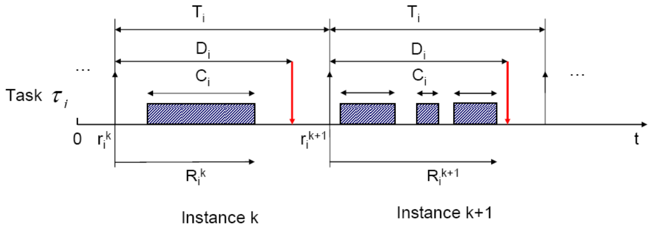


Figure 1. Model

Throughout the paper, we assume that all timing characteristics are non-negative integers, i.e. they are multiples of some elementary time interval (for example the “CPU tick”, the smallest indivisible CPU time unit):

We introduce several notations for a periodic task $\tau_i = (C_i, D_i, T_i)$ used to compute the worst case response time of a task:

- τ_i^k: The k^{th} instance of τ_i

- r_i^1: Release time of the first instance of τ_i
- $r_i^k = r_i^1 + (k-1)T_i$: Release time of τ_i^k
- R_i^k: Response time of τ_i^k released at time r_i^k
- R_i: Worst-case response time of τ_i

3.2. The specific asynchronous scenario

Here we give some interesting properties which are satisfied by the specific asynchronous scenario we propose and which lead to the conclusion that the worst case response time of a task in our asynchronous scenario is obtained for any task for its second release.

In this section we assume that the relative deadline for each task equals its period, i.e. $D_i = T_i$. This assumption will be weakened in section 6.

We first show in lemma 1 that with harmonic asynchronous tasks, two instances belonging to any two tasks can never be released at the same time if their release times are not equal modulo their periods.

Lemma 1: Let $\Gamma_n = \{\tau_1, \tau_2, \cdots, \tau_n\}$ be a system of n independent harmonic (i.e. $T_i \mid T_{i+1}, \forall i \in \{1, \cdots, n-1\}$) preemptive tasks ordered by decreasing priorities ($T_i \leq T_{i+1}, \forall i \in \{1, \cdots, n-1\}$).

If there exist two tasks $\tau_i, \tau_j \in \Gamma_n$, ($i < j$) such that $r_j^1 \neq r_i^1 \bmod[T_i]$ [2], then $\nexists k, l \geq 0$ such that $r_j^k = r_i^l$.

Proof: (by contradiction)

Let us assume that there exist two tasks $\tau_i, \tau_j \in \Gamma_n$, ($i < j$) such that $r_j^1 \neq r_i^1 \bmod[T_i]$, and $\exists k, l \geq 0$ such that $r_j^k = r_i^l$.

$$\begin{array}{lll} r_j^k = r_i^l & \Leftarrow & r_j^1 + (k-1)T_j = r_i^1 + (l-1)T_i \\ & \Leftarrow & r_j^1 = r_i^1 + (l-1)T_i - (k-1)T_j \\ & \Leftarrow & r_j^1 = r_i^1 \bmod[T_i] \text{ as } T_i \mid T_j. \end{array}$$

Contradicts the hypothesis and thus, ends the proof.

□

We now show in theorem 1 that from the point of view of any task in the system, the schedule repeats identically from the second instance.

Theorem 1: (inspired by theorem 2.48 in [24])

Let $\Gamma_n = \{\tau_1, \tau_2, \cdots, \tau_n\}$ be a system of n asynchronous independent periodic preemptive tasks ordered by decreasing priorities ($T_i \leq T_{i+1}, \forall i \in \{1, \cdots, n-1\}$). Let $r_1^1, r_2^1, \cdots, r_n^1$ be respectively the release time of their first instances. Let $(s_i)_{1 \leq i \leq n}$ be the sequence inductively defined by

$$\begin{cases} s_1 = r_1^1 \\ s_i = r_i^1 + \left\lceil \dfrac{(s_{i-1} - r_i^1)^+}{T_i} \right\rceil \cdot T_i \quad \forall i \in \{2, \cdots, n\} \end{cases} \tag{1}$$

Then,
if Γ_n is schedulable up to $s_n + H_n$, with $H_n = LCM(T_1, T_2, \cdots, T_n)$ and $x^+ = max\{x, 0\}$, then Γ_n is schedulable and periodic from s_n with period H_n.

Proof: (By induction on the number of tasks n)

The property is straightforward for the simple case where $n = 1$: indeed, the schedule for task τ_1 is periodic of period T_1 from its first release ($s_1 = r_1^1$) since $C_1 \leq T_1$, otherwise the deadline of the first instance is missed. Let us now assume that the property is true up to $n = i - 1$ and $\Gamma_i = \{\tau_1, \tau_2, \cdots, \tau_i\}$ is schedulable up to $s_i + H_i$, with $H_i = LCM(T_1, T_2, \cdots, T_i)$. Notice that s_i is the first release time of task τ_i after (or at) s_{i-1}. We have $s_i + H_i \geq s_{i-1} + H_{i-1}$ and by induction hypothesis, the subset $\Gamma_{i-1} = \{\tau_1, \tau_2, \cdots, \tau_{i-1}\}$ is schedulable and periodic from s_{i-1} of period H_{i-1}. As tasks are ordered by priority, the instances of the first ones are not changed by the requests of task τ_i and the schedule repeats at time $s_i + LCM(H_{i-1}, T_i) = s_i + H_i$. Consequently, $\Gamma_i = \{\tau_1, \tau_2, \cdots, \tau_i\}$ is schedulable and its schedule repeats from s_i with period H_i.

□

We now characterize the asynchronous scenario we consider in this paper in corollary 1. This leads to providing a simple method for computing the worst response time of each task in section 5 by using corollary 2, and then a pseudo polynomial FC detailed in section 6.1.

Corollary 1: From the point of view of any task τ_i of a schedulable system $\Gamma_n = \{\tau_1, \tau_2, \cdots, \tau_n\}$ ordered by decreasing priorities ($T_i \leq T_{i+1}, \forall i \in \{1, \cdots, n-1\}$) such that $T_i \mid T_{i+1}$ and $r_{i+1}^1 = r_i^1 - C_{i+1}$, the schedule is periodic from the second instance with period $H_i = T_i$.

Proof: (By induction on the index i of the task)

Let us consider a task τ_i of a schedulable system $\Gamma_n = \{\tau_1, \tau_2, \cdots, \tau_n\}$, we assume that $T_i \mid T_{i+1}$ and $r_{i+1}^1 = r_i^1 - C_{i+1}, \quad \forall i \geq 1$. Thanks to the previous theorem, it is sufficient to prove that $s_i - r_i^1 = T_i, \quad \forall i \geq 2$. This is done by induction on i.

The property is straightforward for the simple case where $i = 2$: indeed, as $C_2 \leq T_2$ and $H_2 = LCM(T_1, T_2) = T_2$, the schedule for task τ_2 is periodic of period T_2 from its second release since $s_2 = r_2^1 + \left\lceil \dfrac{(s_1 - r_2^1)^+}{T_2} \right\rceil \cdot T_2 = r_2^1 + \left\lceil \dfrac{C_2}{T_2} \right\rceil \cdot T_2 = r_2^1 + T_2$ is the first release time of task τ_2 after (or at) $s_1 = r_1^1$. Let us now assume that the property is true up to index $i - 1$ and $\Gamma_i = \{\tau_1, \tau_2, \cdots, \tau_i\}$ is schedulable. Thanks to the previous theorem, we have

$$s_i = r_i^1 + \left\lceil \frac{(s_{i-1} - r_i^1)^+}{T_i} \right\rceil \cdot T_i = r_i^1 + \left\lceil \frac{(T_{i-1} + r_{i-1}^1 - r_i^1)^+}{T_i} \right\rceil \cdot T_i$$

by induction hypothesis.

Thus, $s_i = r_i^1 + \left\lceil \dfrac{(T_{i-1} + C_i)^+}{T_i} \right\rceil \cdot T_i$ since $r_{i-1}^1 = r_i^1 + C_i$.

Now, as $0 < T_{i-1} + C_i < T_i$ due to the scenario imposed

[2] Given $a, b, c \in \mathbb{Z}$: $a = b \bmod[c]$ means that there exists $d \in \mathbb{Z}$ such that $a = b + cd$.

to the first instance of each task and the fact that $T_{i-1} \mid T_i$, then we obtain $s_i = r_i^1 + T_i$.

□

Corollary 2: The worst response time R_i of each task τ_i is obtained in the second instance and is equal to that in all instances greater than 2.

Proof:

Immediately follows from corollary 1 and the fact that $R_i^1 = C_i$ by construction, $R_i^k \geq C_i \quad \forall k \geq 1$, and we consider harmonic tasks.

□

4. A new scheduling representation

A direct consequence of corollary 2 leads us to the conclusion that in the case of a valid schedule, i.e. when all deadlines are met for all tasks, the schedule obtained at level i (the resulting schedule of the i tasks with the highest priority) is periodic with the period $T_i = LCM\{T_j | j = 1, \cdots, i\}$ from the second instance. As such, from the point of view of each task, the interval preceeding the second instance necessarily contains the *transient phase*, corresponding to the initial part of the schedule at level i, and the interval starting at date r_i^2 with the length T_i is isomorphic to the *permanent phase* of the schedule at level i, corresponding to the periodic part of the schedule. The transient phase is always finite due to the existence of the permanent phase and the permanent phase repeats indefinitely.

For a system of n periodic harmonic tasks for which there exists a valid schedule, since the permanent phase repeats indefinitely, we introduce a new scheduling representation. This scheduling representation is obtained by graphically using an *oriented circular disk* called *Dameid* with a reference time instant $t_0 = 0$ corresponding to the time reference in the *classical linear representation* or *Gantt Chart*. The positive direction in *Dameid* is the trigonometrical one, i.e. opposite to that of the hands of a watch. The circumference of *Dameid* at level n corresponds to $H_n = LCM\{T_i \mid i = 1, \cdots, n\}$ where T_i means the period of the i^{th} task and n denotes the number of tasks in the system. In *Dameid*, the different release times for each task are unambiguously determined by the value of their first release time relatively to that of other tasks with respect to the reference date $t_0 = 0$, and the ratio $\dfrac{H_n}{T_i}$ for task τ_i. As an example, figure 2 illustrates the release times of each task for a system consisting of 4 periodic harmonic tasks. In this figure, the first release time of task t_1 is -2, while that of task t_3 is 0.

Figure 3 clarifies our idea for the construction of *Dameid* for a given set of harmonic periodic tasks. This figure illustrates, for the same system with 4 tasks (see Figure 2), the correspondence of the release times of each periodic task in *Dameid* relative to the reference date $t_0 = 0$. The main intuition behind this new representation is to reduce the interval of analysis for a system harmonic periodic tasks whatever their first release times are.

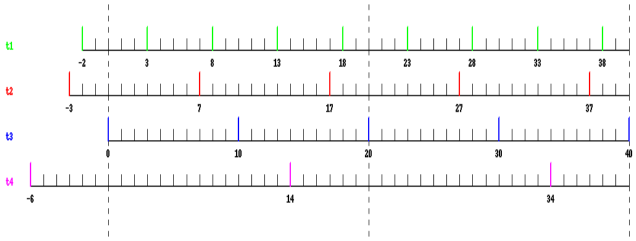


Figure 2. Release times of each task in the classical linear representation or Gantt Chart

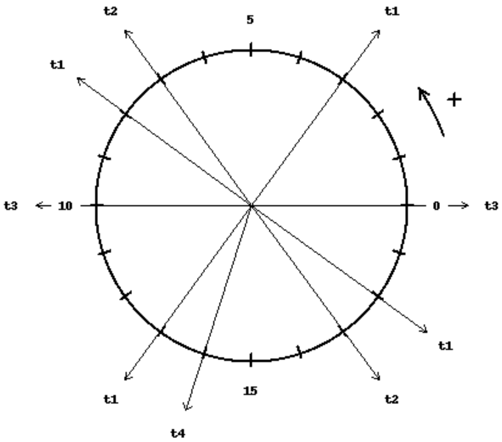


Figure 3. Release times of each task in Dameid

Now, in addition to the release times of each task, let us add the WCETs and explain how *Dameid* can represent schedules.

During the scheduling process from the highest priority task to the lowest priority task, some of the available time units at a given level i, i.e. those which are not executed after the schedule of the first $i-1$ highest priority tasks, are executed by time units corresponding to the WCET C_i of the current task τ_i. This is done in order to obtain the next result for the scheduling analysis of the next task τ_{i+1} with respect to the priorities. As the considered scheduling policy (DM) determines the *total* order in which to perform the scheduling analysis, it follows that the circular representation, i.e. *Dameid*, of circumference corresponding to the *LCM* of periods of all tasks that we have introduced allows us to build directly the *permanent phase* of the system if it is schedulable. Indeed, *Dameid* can be constructed completely independently from the linear representation. In this representation, the WCETs of the tasks correspond to angular sectors, where the angular unit is given by $\dfrac{1}{H_n}$ and

$H_n = LCM\{T_1, T_2, \cdots, T_n\}$.

Figure 6 shows an example of the *Dameid* for the system of which the schedule and the curve of response time as a function of time for each task are illustrated in figure 4 and figure 5. For this system, whose characteristics are summarized in table 1, we assume that task t_1 has a higher priority than task t_2, i.e. tasks are scheduled by using DM. In figure 4, the permanent phase is illustrated by the highlighted zone (blue zone). The curve of the response time of each task according to time (see figure 5) shows that from the time $t = 15$, the response time of each task is constant. We find this result by constructing the *Dameid*. Indeed, the *LCM* of the periods of both tasks t_1 and t_2 is given by $H_2 = LCM(5, 15) = 15$. The release times of task t_1 in *Dameid* with respect to its first release time are given by $r_1^1 = 4$, $r_1^2 = 9$ and $r_1^3 = 14$. For task t_2, we have a single release time equal to $r_2^1 = 0$ because its period $T_2 = H_2 = 15$. Since task t_1 has a higher priority than task t_2, then at each release time of t_1, i.e. at the dates r_1^1, r_1^2 and r_1^3, a sector corresponding to its worst case execution time ($C_1 = 2$ time units) is executed. As task t_2 has a lower priority than task t_1, the filling of the sectors of circumference corresponding to its worst case execution time ($C_2 = 4$ time units) can only be done between the time instants 1 and 4, then time instants 6 and 7. *Dameid* builds the permanent phase of the system directly: in figure 4, task t_2 has two distinct response times, 4 time units for the first activation and 7 time units afterwards, while in the circular representation through *Dameid*, it has a single response time, 7 time units, which corresponds to its response time in the permanent phase.

Tche	r_i^1	C_i	D_i	T_i
t_1	4	2	5	5
t_2	0	4	15	15

Table 1. Characteristics of the tasks

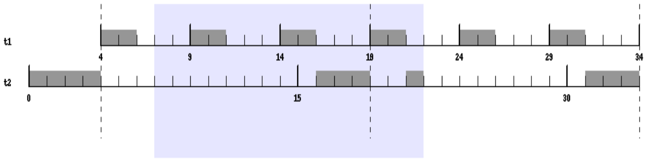


Figure 4. Linear representation / Gantt Chart of the schedule.

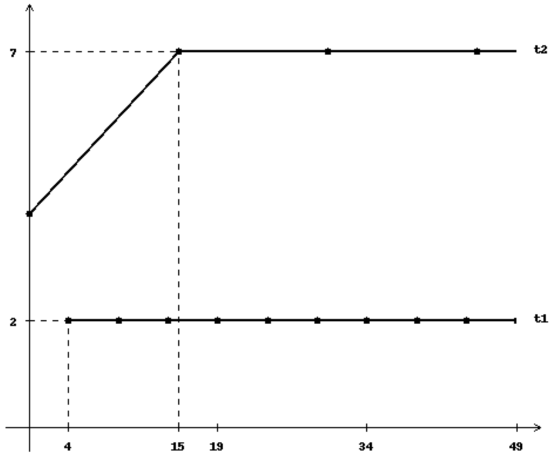


Figure 5. Response time of each task as a function of time.

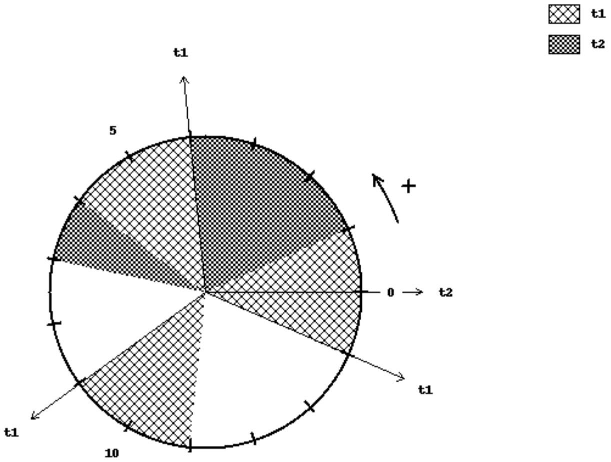


Figure 6. Circular representation of the schedule by using *Dameid*.

This new representation of the schedule is more interesting than the linear representation / Gantt Chart because it is more *compact* and puts greater emphasis on the *available time units* in the resulting schedule. In his thesis ([24]), *Joel Goossens* suggested that the permanent phase is sufficient to guaranteeing the schedulability of a given periodic task set when the cost of preemption is neglected and this permanent phase is directly built by using *Dameid*. We now suppose the asynchronous task set defined in corollary 1 and present the Mesoid approach used to compute the worst case response time of each periodic task.

5. Worst case response time: the Mesoid approach

In this section we provide the method for computing the worst response time of each task in order to check its schedulability. Actually, three classical methods may be used to do so: the utilisation factor of the processor ([25]), the worst response time of each task, or the processor demand ([26]). In this paper we have chosen to use the second approach as it provides a schedulability condition for each task individually. The main idea behind the Mesoid approach is to fill some available time units left by the schedule of higher priority tasks with executed time units corresponding to the execution time of the current task. Since the worst response time is obtained in the second instance w.r.t. corollary 2, we will achieve this goal by applying the method described in [4] to a system where the tasks are not all released simultaneously and where the cost of a preemption is assumed to be zero. This method, unlike those proposed in ([27], [7], [28]), is of lesser complexity since it is not necessary to determine the releases of every task w.r.t. those of higher priority tasks.

As we are in a fixed priority context, the proposed method checks for the schedulability of each task by computing its worst response time, from the task with the highest priority to that with the lowest priority. Hence, from the point of view of any task τ_i of a system $\Gamma_n = \{\tau_1, \tau_2, \cdots, \tau_n\}$ ordered by decreasing priorities ($T_{i-1} \leq T_i, \forall i \in \{2, \cdots, n\}$) such that $T_{i-1}|T_i$ and $r_i^1 = r_{i-1}^1 - C_i$, the elapsed duration between the release of the second instance and the first release r_{i-1}^1 of task τ_{i-1} is given by $T_i - C_i$. Before providing the computation method of the worst case response time, we provide some necessary definitions below.

5.1. Definitions

All the definitions and terminologies used in this section are directly inspired by ([4]) and are applied here to the case of a model where the cost of preemption is assumed to be zero. From the point of view of any task τ_i, the *hyperperiod at level i*, H_i, is given by $H_i = LCM\{T_j\}_{\tau_j \in sp(\tau_i)} = T_i$ as $T_{i-1}|T_i$ for every $i \in \{2, \cdots, n\}$, and $sp(\tau_i)$ is the set of tasks with a period shorter than that of task τ_i. Without any loss of generality we assume that the first task τ_1 starts its execution at time $t = 0$ and that all tasks have different periods. Since at each level the schedule repeats indefinitely from the second instance thanks to corollary 1, it is sufficient to perform the scheduling analysis in the interval $[r_i^1 + T_i, r_i^1 + 2T_i]$ for task τ_i as its response time in its first instance equals its WCET.

We proceed the schedule from the task with the shortest period towards the task with the longest period. Thus, at each level in the scheduling process the goal is to fill available time units in the previous schedule, obtained up to now, with slices of the WCET of the current task, and hence we obtain the next current schedule. Consequently, we represent the previous schedule of every instance τ_i^k of the current task $\tau_i = (C_i, T_i)$ by an ordered set of T_i time units where some have already been executed because of the execution of tasks with shorter periods, and the others are still available for the execution of task τ_i in that instance. We call this ordered set which describes the state of each instance τ_i^k the $\mathcal{M}_i^k$ T_i-*mesoid*. More details on the definition of a T_i-*mesoid* are given in [4]. For the current task $\tau_i = (C_i, T_i)$, there are as many T_i-mesoids as instances. We call $\mathcal{M}_i^{b,2}$ the T_i-*mesoid* corresponding to the second instance of task τ_i **before** being scheduled in the current schedule. The process used to build $\mathcal{M}_i^{b,2}$ for task τ_i will be detailed later in this subsection. Still, from the point of view of task τ_i, we define for the mesoid $\mathcal{M}_i^{b,2}$ the corresponding *universe* X_i^2 to be the ordered set, compatible with that of the mesoid, which consists of all the availabilities of $\mathcal{M}_i^{b,2}$ – that is to say, all the possible values that C_i can take in $\mathcal{M}_i^{b,2}$. Task τ_i will be said to be *potentially schedulable* if and only if

$$C_i \in X_i^2 \quad \forall\, i \in \{1, \cdots, n\} \tag{2}$$

This equation verifies that C_i belongs to the universe at level i. If it does not, then the system is clearly not schedulable. When equation (2) holds for a given task τ_i, we call $\mathcal{M}_i^{a,2}$ the T_i-*mesoids* corresponding to the second instance of task τ_i **after** τ_i has been scheduled. $\mathcal{M}_i^{a,2}$ is a function of $\mathcal{M}_i^{b,2}$ which itself is a function of $\mathcal{M}_{i-1}^{a,2}$, both detailed as follows.

Let f be the function such that $\mathcal{M}_i^{b,2} = f(\mathcal{M}_{i-1}^{a,2})$ which transforms the T_{i-1}-mesoid after task τ_{i-1} has been scheduled at level $i-1$ into the T_i-mesoid before task τ_i is scheduled at level i.

As mentioned in [4], a mesoid consists only of time units already executed denoted by "e" and time units still available denoted by "a". Moreover, the cardinal of a mesoid is equal to the period of the task under consideration whatever the level is. As such, the function f transforms a time unit already executed (resp. still available) in $\mathcal{M}_{i-1}^{a,2}$ into a time unit already executed (resp. still available) in $\mathcal{M}_i^{b,2}$ by following an index ψ which enumerates, according to naturals, the time units (already executed or still available) in $\mathcal{M}_{i-1}^{a,2}$ of task τ_{i-1} after τ_{i-1} has been scheduled. As the elapsed duration between the release of the second instance of task τ_i and the release of the first instance of τ_{i-1} is $T_i - C_i$, then ψ starts from the time unit right after $\gamma_i = T_i - C_i \bmod [T_{i-1}]$ time units in the mesoid $\mathcal{M}_{i-1}^{a,2}$ towards the last time unit, and then circles around to the beginning of the mesoid $\mathcal{M}_{i-1}^{a,2}$ again, until we get the T_i-mesoid $\mathcal{M}_i^{b,2}$. This T_i-mesoid is obtained when $\psi = T_i$. Indeed, the previous schedule at level i (the schedule obtained at level $i-1$) consists of $H_{i-1} = T_{i-1}$ time units whereas the schedule of the current task τ_i is computed upon $H_i = T_i$ time units. Thus, that amounts to extending the previous schedule from T_{i-1} to T_i time units by identically repeating the previous schedule as often as necessary to obtain H_i time units. Due to the particular releases of the first instance of each task, i.e. $r_{i+1}^1 = r_i^1 - C_{i+1}$ $\forall\, i \in \{1, \cdots, n-1\}$, notice that index ψ in contrast to index ζ used in [4] which started from the first time unit, starts from the time unit right after $\gamma_i = T_i - C_i \bmod [T_{i-1}]$ time units in the mesoid $\mathcal{M}_{i-1}^{a,2}$. Since τ_1 is the task with the shortest period, then $sp(\tau_1) = \{\tau_1\}$. Because τ_1 is never preempted, we have $\mathcal{M}_1^{b,2} = \{1, 2, \cdots, T_1\}$ and therefore we obtain $\mathcal{M}_1^{a,2} = \{(C_1), 1, 2, \cdots, T_1 - C_1\}$.

Let g be the function such that $\mathcal{M}_i^{a,2} = g(\mathcal{M}_i^{b,2})$ which transforms the T_i-mesoid $\mathcal{M}_i^{b,2}$ before task τ_i has been scheduled at level i into the T_i-mesoid $\mathcal{M}_i^{a,2}$ after task τ_i has been scheduled at level i.

5.2. Worst case response time with a Mesoid

For the T_i-mesoid $\mathcal{M}_i^{b,2}$, we will compute the response time R_i^2 of task τ_i in the second instance by adding to the WCET C_i all the consumptions appearing in that T_i-mesoid before the availability corresponding to C_i [4]. This yields the worst-case response time R_i of task τ_i since at each level the schedule becomes periodic from the second instance, that is to say $R_i^k = R_i^2 \ \forall\, k \geq 2$, and $R_i^1 = C_i \ \forall\, i \geq 1$.

Now we can build $\mathcal{M}_i^{a,2} = g(\mathcal{M}_i^{b,2})$: function g transforms a time unit already executed in $\mathcal{M}_i^{b,2}$ into a time unit already executed in $\mathcal{M}_i^{a,2}$, and transforms a time unit still available into either a time unit still available or a time unit already executed w.r.t. the following condition. We use an index which enumerates according to numerals the time units in $\mathcal{M}_i^{b,2}$ from the first to the last one, at each step in the incremental process, if the current value of the index is less than or equal to R_i^2, function g transforms the time unit still available into a time unit already executed due to the execution of instance τ_i^2, otherwise g transforms it into a time unit still available. Indeed, function g fills available time units in the current schedule with slices of the WCET in each T_i-mesoid, leading to the previous schedule for the next task at level $i + 1$ w.r.t. priorities. To summarize, for every task τ_i, we have

$$\tau_i : \begin{cases} \mathcal{M}_i^{b,2} : T_i\text{-mesoid before } \tau_i \text{ is scheduled at level } i \\ \\ \mathcal{M}_i^{a,2} : T_i\text{-mesoid after } \tau_i \text{ is scheduled at level } i. \end{cases}$$

6. Deadline reduction factor

6.1. Worst case response time computation

The approach proposed here leads to a new schedulability condition for harmonic hard real-time systems. This condition is new in the sense that in addition to providing a necessary and sufficient schedulability condition, it also reduces the feasibility interval for a given harmonic asynchronous system.

In the scheduling process, at each level i, the basic idea consists in filling availabilities in the mesoid $\mathcal{M}_i^{b,2}$ before task τ_i is scheduled, with slices of its WCET. This is why it is fundamental to calculate the corresponding response time. This yields the worst case response time and allows us to conclude on the schedulability of task τ_i w.r.t. priorities. In the case where τ_i is schedulable, we build $\mathcal{M}_i^{a,2}$, after τ_i has been scheduled, in order to check the schedulability of the next task, and so on, otherwise the system is not schedulable. Thanks to everything we have presented up to now, τ_1 is scheduled first and $r_1^1 = 0$. The latter statement implies that **before** τ_1 is scheduled, its WCET can potentially take any value from 1 up to the value of its period T_1. Since task τ_1 is never preempted, then $\mathcal{M}_1^{b,2} = \{1, 2, \cdots, T_1\}$ and $X_1^2 = \{1, 2, \cdots, T_1\}$. Moreover, its response time is also equal to C_1. Consequently, the corresponding T_1-*mesoids* associated to task τ_1 are given by

$$\tau_1 : \begin{cases} \mathcal{M}_1^{b,2} = \{1, 2, \cdots, T_1\} \\ \\ \mathcal{M}_1^{a,2} = \{(C_1), 1, 2, \cdots, T_1 - C_1\} \end{cases}$$

We assume that the first $i - 1$ tasks with $2 \leq i \leq n$ have already been scheduled, i.e. the T_{i-1}-mesoid $\mathcal{M}_{i-1}^{a,2}$ of task τ_{i-1} is known, and that we are about to schedule task τ_i.

As explained in the previous section, the T_i-mesoid $\mathcal{M}_i^{b,2} = f(\mathcal{M}_{i-1}^{a,2})$ of task τ_i is built thanks to index ψ on $\mathcal{M}_{i-1}^{a,2}$ of task τ_{i-1} without forgetting to start from the time unit right after $\gamma_i = T_i - C_i \operatorname{mod}[T_{i-1}]$ time units rather than the first time unit as in [4]. Again this is due to the particular release of the first instances of tasks: $r_i^1 = r_{i-1}^1 - C_i$. We can therefore determine the universe X_i^2 when the T_{i-1}-mesoid $\mathcal{M}_{i-1}^{a,2}$ is known. Unless the system is not schedulable, i.e. $C_i \notin X_i^2$, we assume that task τ_i is potentially schedulable, i.e. $C_i \in X_i^2$. The response time R_i^2 of task τ_i in its k^{th} instance (with $k \geq 2$), i.e. in the k^{th} T_i-*mesoid* will be obtained by summing C_i with all consumptions prior to C_i in the corresponding mesoid. The worst-case response time R_i of task τ_i will then be given by

$$R_i = R_i^2$$

This equation leads us to say that task τ_i is schedulable if and only if

$$R_i \leq T_i \qquad (3)$$

If for task τ_i expression (3) holds, then $\mathcal{M}_i^{a,2} = g(\mathcal{M}_i^{b,2})$ will be deduced as explained in the previous section. For the sake of clarity, whenever there are two consecutive consumptions in a *mesoid*, this amounts to considering only one consumption which is the sum of the previous consumptions. That is to say that after determining the response time of task τ_i in its k^{th} mesoid, if $\mathcal{M}_i^{a,k} = \{(c_1), (c_2), 1, 2, \cdots\}$, then this is equivalent to $\mathcal{M}_i^{a,k} = \{(c_1 + c_2), 1, 2, \cdots\}$.

Below, we present our scheduling algorithm which, for a given task, on the one hand first determines the value of $\gamma_i = T_i - C_i \bmod [T_{i-1}]$ relative to priorities, then, on the other hand the schedulability condition. Recall that the elapsed duration between the release of the second instance and the first release is $T_i - C_i$. The scheduling algorithm has the following nine steps. Since the task with the shortest period, namely task τ_1, is never preempted, the loop starts from the index of the task with the second shortest period, namely task τ_2 as the schedule proceeds towards tasks with longer periods.

```
1: for $i = 2$ to $n$ do
2:     Determine the release time of the first instance of task $\tau_i$:
           $r_i^1 = r_{i-1}^1 - C_i$
       and compute $\gamma_i = T_i - C_i \bmod [T_{i-1}]$ of the second instance of $\tau_i$ w.r.t. $\tau_{i-1}$.
3:     Build the $T_i$-mesoid $\mathcal{M}_i^{b,2} = f(\mathcal{M}_{i-1}^{a,2})$ of task $\tau_i$ before it is scheduled. This construction is based on a modulo $T_i$ arithmetic using index $\psi$ on $\mathcal{M}_{i-1}^{a,2}$ without forgetting to start from the time unit right after $\gamma_i = T_i - C_i \bmod [T_{i-1}]$ time units rather than the first time unit as in [4]. This is due to the particular release of tasks.
4:     For the $T_i$-mesoid $\mathcal{M}_i^{b,2}$ resulting from the previous step, build the corresponding universe $X_i^2$ which consists of the ordered set of all availabilities of $\mathcal{M}_i^{b,2}$. Notice that this set corresponds to the set of all possible values that the WCET $C_i$ of task $\tau_i$ can take in $\mathcal{M}_i^{b,2}$.
5:     Since $\tau_i$ is potentially schedulable, i.e. its WCET $C_i \in X_i^2$, we must verify that it is actually schedulable. Clearly, if $C_i \notin X_i^2$, then task $\tau_i$ is not schedulable because the deadline of the task is exceeded.
6:     Determine the response time $R_i^k$ of task $\tau_i$ in its $k^{th}$ instance, i.e. in the $k^{th}$ $T_i$-mesoid. This is obtained by summing $C_i$ with all the consumptions prior to $C_i$ in the corresponding mesoid. Deduce the worst-case response time $R_i$ of task $\tau_i$.
           $R_i = R_i^2$
       It is worth noticing that task $\tau_i$ is schedulable if and only if
           $R_i \leq D_i.$
7:     If $R_i \leq D_i$, then build $\mathcal{M}_i^{a,2} = g(\mathcal{M}_i^{b,2})$, increment $i$, and go back to step 2 as long as there remain potentially schedulable tasks in the system.
8:     If $R_i > D_i$, then the system $\{\tau_i = (C_i, T_i)\}_{1 \leq i \leq n}$ is not schedulable.
9: end for
```

Thanks to the above algorithm, a system of n tasks $\{\tau_i = (C_i, T_i)\}_{1 \leq i \leq n}$, with harmonic periods and first released such that $r_i^1 = r_{i-1}^1 - C_i$, is schedulable if and only if

$$R_i = R_2^2 \leq D_i \qquad \forall i \in \{1, 2, \cdots, n\} \tag{4}$$

6.2. Computation of α

The value of α is given by: $\alpha = \max_{1 \leq i \leq n} \left(\dfrac{R_i}{T_i}\right)$.

This value of α guarantees that no task fails at run-time. We recall that for the synchronous scenario, the worst case response time of task τ_i is given by:

$$R_i = C_i + \sum_{j \in hp(i)} \left\lceil \frac{R_i}{T_j} \right\rceil C_j$$

Example

Let us consider $\{\tau_1,\ \tau_2,\ \tau_3,\ \tau_4\}$ to be a system of four tasks with harmonic periods and first released such that $r_i^1 = r_{i-1}^1 - C_i$. The characteristics are defined in table 2.

Table 2. Characteristics of the tasks

	C_i	T_i
τ_1	2	5
τ_2	4	15
τ_3	5	30
τ_4	7	60

The shorter the period of a task is, the higher its level is. Thus, as depicted in table 2, τ_1 has the highest level and task τ_4 the lowest level. Thanks to our scheduling algorithm, for task τ_1 whose first release time is $r_1^1 = 0$, we have

$$\tau_1 : \begin{cases} \mathcal{M}_1^{b,2} = \{1, 2, 3, 4, 5\} \\ R_1 = 2 \\ \mathcal{M}_1^{a,2} = \{(2), 1, 2, 3\} \end{cases}$$

$\gamma_2 = T_2 - C_2 \bmod [T_1] = 15 - 4 \bmod [5] = 1$, thus for task τ_2 whose first release time is $r_2^1 = r_1^1 - C_2 = -4$, we have

$$\tau_2 : \begin{cases} \mathcal{M}_2^{b,2} = \{(1), 1, 2, 3, (2), 4, 5, 6, (2), 7, 8, 9, (1)\} \\ R_2 = 4 + 2 + 1 = 7 \\ \mathcal{M}_2^{a,2} = \{(7), 1, 2, (2), 3, 4, 5, (1)\} \end{cases}$$

$\gamma_3 = T_3 - C_3 \bmod [T_2] = 30 - 5 \bmod [15] = 10$, thus for task τ_3 whose first release time is $r_3^1 = r_2^1 - C_3 = -4 - 5 = -9$, we have

$$\tau_3 : \begin{cases} \mathcal{M}_3^{b,2} = \{(1), 1, 2, 3, (8), 4, 5, (2), 6, 7, 8, (8), 9, 10, (1)\} \\ R_3 = 5 + 8 + 1 = 14 \\ \mathcal{M}_3^{a,2} = \{(16), 1, 2, 3, (8), 4, 5, (1)\} \end{cases}$$

$\gamma_4 = T_4 - C_4 \bmod [T_3] = 60 - 7 \bmod [30] = 23$, thus for task τ_4 whose first release time is $r_4^1 = r_3^1 - C_4 = -9 - 7 = -16$, we have

$$\tau_4 : \begin{cases} \mathcal{M}_4^{b,2} = \{(4), 1, 2, (17), 3, 4, 5, (8), 6, 7, (17), 8, 9, 10, (4)\} \\ R_4 = 7 + 8 + 17 + 4 = 36 \\ \mathcal{M}_4^{a,2} = \{(53), 1, 2, 3, (4)\} \end{cases}$$

Consequently, the set of tasks $\{\tau_1, \tau_2, \tau_3, \tau_4\}$ with harmonic periods and first released such that $r_i^1 = r_{i-1}^1 - C_i$ is schedulable. The schedule with the above characteristics

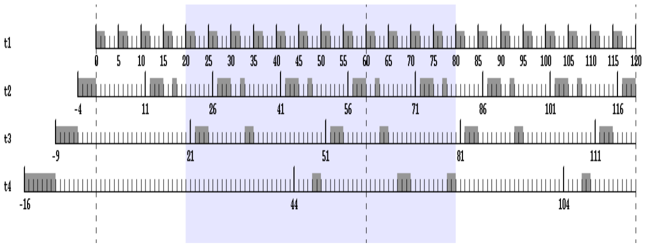


Figure 7. Execution of a set of harmonic tasks with $r_i^1 = r_{i-1}^1 - C_i, \ \forall \, i \in \{2, \cdots, 4\}$

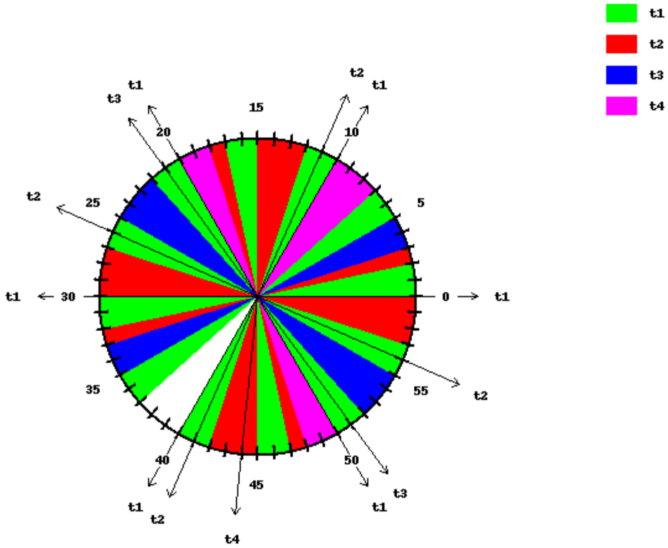


Figure 8. Circular representation of the schedule for a set of harmonic tasks with $r_i^1 = r_{i-1}^1 - C_i, \forall i \in \{2, \cdots, 4\}$

is depicted in figure 7 and the circular representation of the schedule by using *Dameid* is depicted in figure 8.
The schedule of the same set of tasks released simultaneously is depicted in figure 9 and the circular representation of the schedule by using *Dameid* is depicted in figure 10.

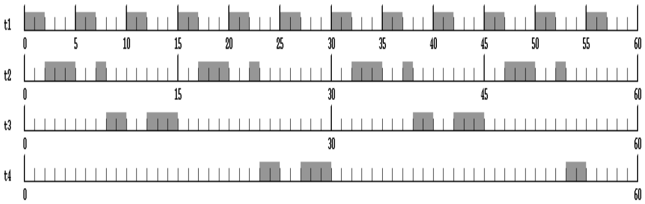


Figure 9. Execution of a set of harmonic tasks with $r_i^1 = 0 \ \forall \, i \in \{1, \cdots, 4\}$

It is worth noticing here the large variation between the two scenarios in terms of the tasks' response times. In fact, the worst case response time of task τ_4 in figure 7 and figure 8 is 36 time units whereas it is 55 time units in figure 9 and figure 10. This phenomenon is even more apparent in the next section with the experimental results where we gradually and uniformly decrease the value of the relative deadlines for all tasks by the same factor to highlight the advantage of our approach.

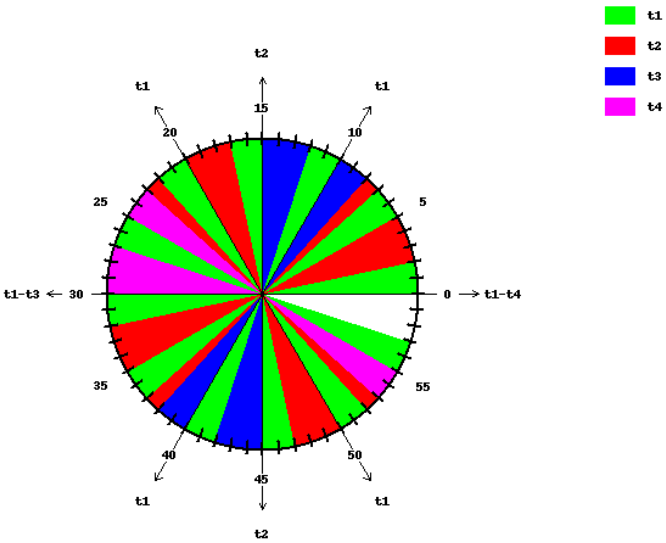


Figure 10. Circular representation of the schedule for a set of harmonic tasks with $r_i^1 = 0 \ \forall \, i \in \{1, \cdots, 4\}$

Tasks	$R_i^{synchronous}$	$R_i^{asynchronous}$
τ_1	2	2
τ_2	8	7
τ_3	15	14
τ_4	55	36

This leads us to obtain $\alpha^{synchrnous} = max(2/5, 8/15, 15/30, 55/60) = 0.91$ whereas $\alpha^{asynchrnous} = max(2/5, 7/15, 14/30, 36/60) = 0.60$, which means the improvement performed in this case is of 34.54%

7. Experimental results

In this section we present some experimental results found by using the approach we have developed above. To achieve these experimental results, we proceed in two steps. First, we compare the minimum deadline reduction factor α obtained in the synchronous scenario with that obtained in our specific asynchronous scenario. Second, we extend this comparison concerning the value α to the value of α obtained for an arbitrarily generated scenario of the first release times for all tasks. This extension is performed by using more extensive experiments in order to get more accurate conclusions with regard to the contributions of the proposed approach. As in ([1]), we consider a set of harmonic tasks scheduled with the Deadline Monotonic algorithm.

The first step in our process of comparing the value of α for given scenarii of first release for all tasks consists in performing 10000 experiments for each graph, where every task set consists of $n = 10$ harmonic tasks. The total utilization factor of the processor is randomly chosen between 0.7 and 1 for each task set. Hence, we can evaluate the gain of our specific asynchronous scenario defined in corollary 1 in section 3, compared to the synchronous one. We set $\alpha = \frac{D_i}{T_i}$, and we gradually and uniformly decrease the value of the relative deadlines D_i by the same factor for all tasks in each set. In both the synchronous and the asynchronous scenario, we plot the curves corresponding to the smallest value of α, as a function of the total utilization factor of the processor, for the task set to remain schedulable. The resulting graphic is displayed in figure 11. If the value of α is denoted $\alpha^{synchronous}$ in the synchronous scenario and $\alpha^{asynchronous}$ in our asynchronous scenario, the gain can be computed as follows:

$$gain = \frac{\alpha^{synchronous} - \alpha^{asynchronous}}{\alpha^{synchronous}} \times 100$$

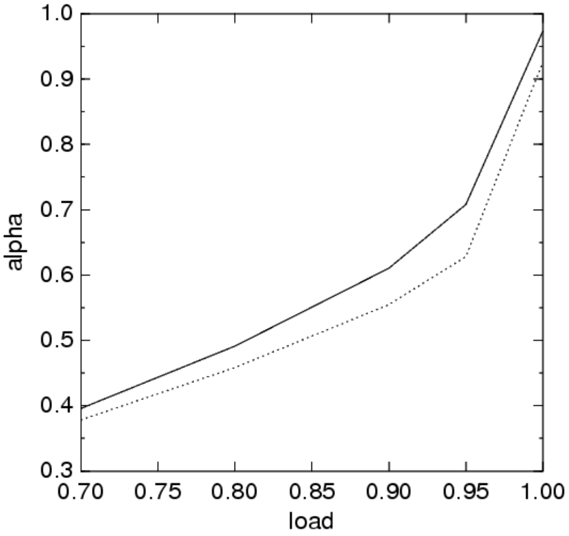


Figure 11. Value of α with our asynchronous scenario and with the synchronous scenario

In figure 11, the solid curve represents the result obtained for α in our specific asynchronous case whereas the dotted curve represents the result obtained in the synchronous case. In both cases, we start with a schedulable task set $\forall \tau_i, D_i = T_i$. From [20], $U \leq 1$ is a necessary and sufficient condition for the schedulability of a harmonic task set as tasks are scheduled with DM, equivalent to RM when $\forall \tau_i, D_i = T_i$. We can see that for a small load, we obtain almost the same α both in the synchronous and in the specific asynchronous cases.

Concerning the second step in our process of comparing the value of α for given scenarii of first release times for all tasks, we perform twice as many experiments than for the first step. That is to say, we perform 20000 experiments for each graph, and every task set still consists of $n = 10$ harmonic tasks. Again, the total utilization factor of the processor is randomly chosen between 0.7 and 1 for each task set. As such, we can evaluate the gain of α obtained in our specific asynchronous scenario, compared to that obtained in the synchronous scenario on the one hand, and to the mean value obtained for a set of arbitrarily generated scenarii on the other hand. As for the first step, we set $\alpha = \frac{D_i}{T_i}$, and we gradually and uniformly decrease the value of the relative deadlines D_i by the same factor for all tasks in each set. For the synchronous, and the asynchronous scenarii, we plot the curves corresponding to the smallest value of α. For the set of arbitrarily generated scenarii, we plot the curves corresponding to the mean value of α. This is performed in each case as a function of the total utilization factor of the processor, for the task set to remain schedulable. The curves obtained are displayed in figure 12.

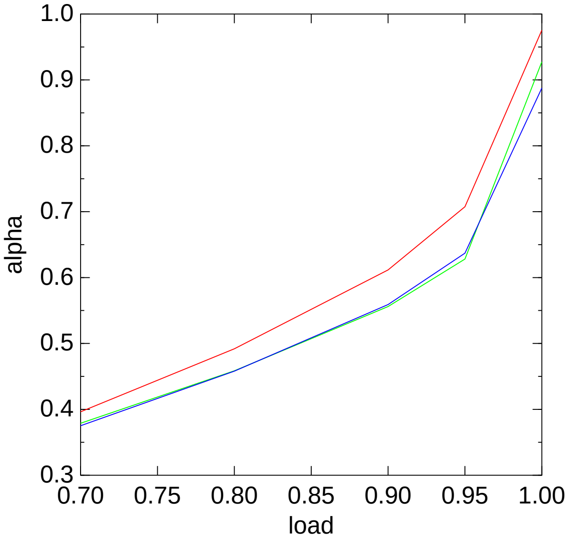


Figure 12. Value of α with our asynchronous scenario, then with the synchronous scenario and the mean of a set of arbitrarily generated scenarii

In figure 12, the curve in *red* represents the result obtained for α by using our specific asynchronous scenario. The curve in *green* represents the result obtained for the synchronous case and the curve in *blue* represents the mean value obtained for a set of arbitrarily generated scenarii. In all the cases, we start with a schedulable task set $\forall \tau_i, D_i = T_i$ and $U \leq 1$ remains a necessary and sufficient condition for the schedulability of a harmonic task set as tasks are scheduled with DM. It is worth noticing that DM is equivalent to RM when $\forall \tau_i, D_i = T_i$.

We can see that we always obtain almost the same value for α both in the synchronous case and for the mean value obtained for a set of arbitrarily generated scenarii.

For a small load, the value of α varies very slightly whatever the scenario of first release for all tasks is. In both steps, this is due to the fact that with a small load the worst case response times of the tasks are less influenced by the first release times of other tasks. When the load increases, the gain also increases, reaches and remains at a maximum of 14.3% for $U = 0.95$. Over the load $U = 0.95$, the gain steadily decreases when U tends to 1 and α tends to 1. At high loads, the worst case response time of a task tends to its period and thus α tends to 1. In this latter case, the improvement obtained with our spacific asynchronous scenario becomes less significant.

8. Conclusion

In this paper we have proposed a new approach for a better control of periodic tasks scheduled with Deadline Monotonic scheduling algorithm. We have considered a specific asynchronous task set and harmonic tasks that enables us to significantly reduce the worst case response time of each task thus reducing the jitter of each task for a better control. The asynchronous scenario we considered makes it possible to significantly reduce the complexity of the worst case response time computation. We have then considered the Mesoid approach to compute the worst case response time of a task in an asynchronous scenario. We have used the Mesoid approach to compute the minimum deadline reduction factor characterizing the benefit in terms of worst case response time reduction. We have proved by extensive simulations that the gain in terms of deadline reduction can reach 14.3% with our particular asynchronous scenario compared to the synchronous scenario and to an arbitrarily generated scenario. This makes it possible to better control the jitter of the tasks when considering control loops. Future work will compare the deadline reduction factor obtained with EDF with the one we have obtained with our specific asynchronous scenario.

References

[1] P. Meumeu Yomsi, L. George, Y. Sorel, and D. De Rauglaudre. Improving the Sensitivity of Deadlines with a Specific Asynchronous Scenario for Harmonic Periodic Tasks scheduled by FP. *The Fourth International Conference on Systems (ICONS'09),* Cancun, Mexico, March 1 - 6 2009.

[2] Giorgio Buttazzo Enrico Bini, Marco Di Natale. Sensitivity Analysis for Fixed-Priority Real-Time Systems. *Proceedings of the 18th Euromicro Conference on Real-Time Systems (ECRTS'06),* Dresden, Germany July 5-7, 2006.

[3] Ismael Ripoll Patricia Balbastre and Alfons Crespo. Optimal deadline assignment for periodic real-time tasks in dynamic priority systems. *Proceedings of the 18th Euromicro Conference on Real-Time Systems (ECRTS'06),* Dresden, Germany July 5-7, 2006.

[4] P. Meumeu Yomsi and Sorel Y. Extending Rate Monotonic Analysis with Exact Cost of Preemptions for Hard Real-Time Systems. *Proceedings of 19th Euromicro Conference on Real-Time Systems, ECRTS'07,* Pisa, Italy, Jul. 2007.

[5] S. Baruah, R. Howell, and L. Rosier. Algorithms and complexity concerning the preemptive scheduling of periodic real-time tasks on one processor. *Real-Time Systems*, Vol. 2, pp. 301-324, 1990.

[6] K. Tindell, A. Burns, and A. J. Wellings. Analysis of hard real-time communications. *Real-Time Systems*, Vol. 9, pp. 147-171, 1995.

[7] L. George, N. Rivierre, and M. Spuri. Preemptive and non-preemptive scheduling real-time uniprocessor scheduling. *INRIA Research Report*, No. 2966, September 1996.

[8] E. Bini and G. Buttazzo. The Space of EDF Feasible Deadlines. *Proceedings of the 19th Euromicro Conference on Real-Time Systems (ECRTS'07),* Pisa, Italy, July 4-6 2007.

[9] S. Baruah, G. Buttazo, S. Gorinsky, and G. Lipari. Scheduling periodic task systems to minimize output jitter. *In* 6^{th} *Conference on Real-Time Computing Systems and Applications*, pp. 62-69, 1999.

[10] A. Cervin, B. Lincoln, J. Eker, K. Arzen, and Buttazzo G. The jitter margin and its application in the design of real-time control systems. *In proceedings of the IEEE Conference on Real-Time and Embedded Computing Systems and Applications*, 2004.

[11] D. Marinca, P. Minet, and L. George. Analysis of deadline assignment methods in distributed real-time systems. *Computer Communications*, Elsevier, To appear, 2004.

[12] M. Joseph and P. Pandya. Finding response times in a real-time system. *BCS Comp. Jour.*, 29(5), pp. 390-395,, 1986.

[13] M. Grenier, J. Goossens, and N. Navet. Near-optimal fixed priority preemptive scheduling of offset free systems. *Proc. of the 14th International Conference on Network and Systems (RTNS'2006)*, Poitiers, France, May 30-31, 2006 2006.

[14] Giorgio Buttazzo Enrico Bini. Schedulability Analysis of Periodic Fixed Priority Systems. *IEEE Transactions On Computers, Vol. 53, No. 11*, Nov.2004.

[15] J.P. Lehoczky. Fixed priority scheduling of periodic task sets with arbitrary deadlines. *Proceedings 11th IEEE Real-Time Systems Symposium*, pp 201-209, Dec. Lake Buena Vista, FL, USA, 1990.

[16] J. Y. T. Leung and M.L. Merril. A note on premptive scheduling of periodic, Real Time Tasks. *Information Processing Letters*, Vol 11, num 3, Nov. 19980.

[17] Annie Choquet-Geniet and Emmanuel Grolleau. Minimal schedulability interval for real-time systems of periodic tasks with offsets. *Theor. Comput. Sci.*, 310(1-3):117–134, 2004.

[18] N. C. Audsley. Optimal priority assignment and feasibility of static priority tasks with arbitrary start times. *Dept. Comp. Science Report YCS 164, University of York*, 1991.

[19] J. Goossens. Scheduling of offset free systems. *Real-Time Systems*, 24(2):239-258, March 2003.

[20] G. C. Buttazzo. Rate Monotonic vs. EDF: Judgment Day. *Real-Time Systems, 29, 5-26,* 2005.

[21] E. Bini and A. Cervin. Delay-Aware Period Assignment in Control Systems. *Proceedings of the 26th IEEE International Real-Time Systems Symposium (RTSS'08),* Barcelona, Spain, Nov. 30 to Dec. 3 2008.

[22] S. Baruah, A. K. Mok, and L. Rosier. Preemptively scheduling hard real-time sporadic tasks on one processor. *Proceedings of the 11th Real-Time Systems Symposium*, pp. 182-190, 1990.

[23] L. C. Liu and W. Layland. Scheduling algorithms for multiprogramming in a hard real time environment. *Journal of ACM*, Vol. 20, No 1, pp. 46-61, January 1973.

[24] J. Goossens. *Scheduling of Hard Real-Time Periodic Systems with Various Kinds of Deadline and Offset Constraints.* PhD thesis, Université Libre de Bruxelles, 1998.

[25] C.L. Liu and J.W. Layland. Scheduling algorithms for multiprogramming in a hard-real-time environment. *Journal of the ACM*, 1973.

[26] A.K. Mok S.K. Baruah and L.E. Rosier. Preemptively scheduling hard realtime sporadic tasks on one processor. *In proc. 11th IEEE Real-Time Systems Symposium*, 1990.

[27] Joseph Y.-T. Leung and M. L. Merrill. A note on preemptive scheduling of periodic, real-time tasks. *Information Processing Letters*, 1980.

[28] J. Leung and Whitehead J. On the complexity of fixed-priority scheduling of periodic real-time tasks. *Performance Evaluation(4)*, 1982.

Searching similar clusters of polyhedra in crystallographic databases

Hans-Joachim Klein and Christian Mennerich
Institut für Informatik
Universität Kiel, 24098 Kiel, Germany
{hjk,chm}@is.informatik.uni-kiel.de

Abstract

A graph-based method is described for searching and ranking clusters of polyhedra in large crystallographic databases. It is shown how topologically equivalent substructures can be determined for a given target cluster based upon a graph representation of polyhedral networks. A mathematical modeling of geometric embeddings of polyhedra graphs is provided which can be used to define geometric similarity of polyhedral clusters. For a special kind of similarity, an algorithm for solving the problem of absolute orientation is applied in order to rank topologically equivalent clusters appropriately.

Keywords: Crystallographic databases, polyhedral clusters, polyhedra graphs, similarity search, ranking

1 Introduction

In recent years, large databases have been built in organic as well as in inorganic chemistry [2], [3]. These systems offer query facilities for searching compounds given publication data, kinds of atoms, symmetry information, etc. For databases storing information on organic and metal-organic crystal structures, it is also possible to search for certain patterns of combinations of atoms [4]. Similarity searching in general has received considerable attention in the field of molecular structures modeled as simple undirected graphs [5]. Rigid substructures can be distinguished and used to build indexes for fast search.

At present, such a kind of searching at the level of substructures is not offered for inorganic crystallographic databases. Whereas for organic compounds, search can be built upon a set of substructures of reasonable size this approach is less meaningful for inorganic compounds. Here a large variety of chemical elements and patterns can be observed. The symmetry is generally higher than in organic and many inorganic molecules leading to a rich diversity of geometric configurations. Local arrangements of atoms play an important role for the description and understanding of structures.

In order to deal with this situation, an approach has been presented in [6] which is based upon a description of inorganic crystal structures at the level of coordination polyhedra. Infinite networks formed by connections of polyhedra can be represented by finite periodic graphs. This modeling allows to build an indexation of polyhedral networks by chains, which can be used for the efficient determination of topologically equivalent substructures. These structures can have a quite different geometry. Hence a method is needed to check for geometric similarity.

It has been argued that to determine geometric transformations first and then to test for preservation of topology is more efficient in connection with geometric graph isomorphism [7]. However, when using rotations, translations, and scaling to investigate geometric isomorphism the problem arises that substructures with great similarity up to a sharp difference at a single position cannot be found in principle.

In order to be flexible with respect to the definition of similarity and to allow the user to decide which differences in the geometry are tolerable, a two-step approach for determining similar structures is applied. In the first step, all substructures in a given set of model structures are determined, which are candidates for the result because they are topologically equivalent to the given search structure. By taking the symmetries of structures into account, the set of candidates can be reduced to symmetrically non-equivalent substructures. In the second step, geometric similarity is checked for all candidates resulting in a ranking. This ranking can be used for presenting the search results. Furthermore, the concrete values of the similarity test provide information about the relationship between the structures.

This paper is an extended version of [1]. It is organized as follows. We start with describing some methods for determining coordination polyhedra as they are implemented in our system. Then we review the graph representation of clusters of polyhedra and the definition of topological equivalence. We discuss some properties of the ordered face representation of polyhedra and show how the embedding of subgraphs in model graphs can be done quite efficiently in many cases. In the fourth section, the problem of geometric similarity is discussed more generally. A modeling of polyhedral clusters as joint structures is developed and some forms of similarity based upon this modeling are proposed. In Section 5, for one such form which is based upon point sets the implementation in our system POLYSEARCH is described and some results are presented. We conclude with remarks concerning the usage of the system and future work.

2 Graph representation and topological equivalence of polyhedral clusters

Graphs are often used to describe structural aspects of chemical compounds. In this section we show how the bonding between atoms in crystals can be represented by special forms of labeled graphs.

2.1 Coordination polyhedra

Inorganic crystal structures are often modeled using coordination polyhedra as components [8]. The vertices of these convex polyhedra represent the atoms which are considered as ligands of a fixed central atom. In case of small sets of ligands and regular forms (e.g. tetrahedra in silicates) the determination of coordination polyhedra is rather straightforward. If the number of ligands and their distance to the central atom grows, polyhedra may become deformed and their determination is less obvious. A generally applicable formal definition of coordination polyhedra in crystal structures seems to be impossible. It is therefore advisable to apply different methods for determining coordination polyhedra and to have a closer look at the results if they show differences. The following methods seem to be suitable into that regard and are offered by our system POLYSEARCH [9]:

- *Search within a given maximal distance.*

All atoms within the given distance are considered as ligands. It is checked whether they determine a convex polyhedron.

- *Determination of the maximal convex hull.*

A convex hull algorithm is used in order to determine the coordination polyhedron (the gift wrapping algorithm [10] is appropriate since the number of vertices of coordination polyhedra is small).

- *Determination of the maximal convex hull restricted by a given number of atoms.*

The convex hull algorithm stops when the given number of atoms is reached.

- *Search for a polyhedron with a specified number n of ligands.*

Sets of n atoms around the central atom are built and checked for convexity. Atoms not contained in such a set may not have a distance to the central atom smaller than the distance to the central atom of any atom of the set.

- *Search for a maximal gap (linear).*

Circumscribing spheres are defined by specifying an upper limit ϵ for the differences between the distances of neighbouring atoms to the central atom. Atoms a, a' belong to the same sphere if there is a sequence $a = a_1, ..., a_n = a'$ such that the distances of a_i and a_{i+1} to the central atom are less than or equal to ϵ for $i = 1, ..., n-1$. A gap between two spheres S, S' with S' circumscribing S is the difference between the minimal distance of an atom in S' and the maximal distance of an atom in S. The gap Δ between S and S' is maximal if all gaps between spheres circumscribed by S and the gap between S' and the next sphere are smaller than Δ.

- *Search for a maximal gap (volume).*

A similar proceeding as in the linear case is applied. Instead of the differences of distances the differences of the volumes of the spheres determined by the gaps are considered.

The linear gap method is often used in the literature [11] and applied in systems such as Pearson's Crystal Data [12] for determining coordination polyhedra. Because of problems which may arise in determining the maximal gap, Pearson's Crystal Data also offers a maximal convex hull algorithm. The volume method seems to be more appropriate in connection with the use of distances between atoms if no assumptions can be made about the arrangement of ligands around the central atom in space.

Further conditions for the search of coordination polyhedra in a given structure can be the kind of central atoms and the kind of ligands. If only homogeneous polyhedra are of interest, for example, a single kind of atoms can be allowed as ligands.

In many structures coordination polyhedra with a small number of ligands (up to six) in near neighbourhood to the central atom are quite regular and have the symmetries of the corresponding ideal forms (tetrahedra, octahedra). Such regularity can, however, often not be found if the number of ligands and their distance to the central atom increases. Then deformations are frequent and the problem of how to measure polyhedral distortion arises [13]. POLYSEARCH compares a polyhedron found by one of the methods presented above with the description of a set of ideal polyhedra. This comparison is done on the basis of the adjacency matrices of the graphs representing the adjacency of the ligands.

For the representation of polyhedra and clusters of polyhedra, a graph form has been introduced in [6]. It uses a unique numbering of vertices and a description of faces by ordered sequences of numbers for coordination polyhedra. Depending on the purpose of their usage, these graphs can be augmented by coordinates to get a complete description of the geometry of clusters or they can be reduced to pure topological information. Two views of coordination polyhedra are distinguished in [6] using the well-known correspondence between convex polyhedra and three-connected planar graphs (Theorem of Steinitz).

Definition: The *geometrical view* of a coordination polyhedron P is a vertex-labeled simple three-connected planar graph $(V \cup \{c\}, E, pos)$. The vertex set $V \cup \{c\}$ represents the ligands of P and the central atom, respectively; the set of edges is determined by the adjacency relationship of the ligands of P and the function $pos : V \cup \{c\} \to At \times \mathbb{R}^3$ assigns to every element of the vertex set the element symbol and the coordinates of the corresponding atom of P.

Information on the symmetries of the polyhedron is not included in the definition since it may be derived from V and pos.

In the first step of our similarity check we look for all clusters in a given set of model structures which are topologically equivalent to a given search cluster. For this search, position data of atoms and the kind of atoms involved are not needed. Hence the following view of polyhedra is suitable:

Definition: The *pure topological view* of a coordination polyhedron P is a simple three-connected planar graph (V, E) with V representing the ligands of P and E the adjacency relationship of the ligands.

Since there is no information on locations of atoms, the central atom must not be represented by a vertex. The topological view of a polyhedron provides no information on its symmetry group. If symmetry properties are of interest, the symmetry group of the polyhedron or subgroups of it may be added.

When polyhedra are considered as elements of clusters of polyhedra some of their vertices become fixed as connecting points. In case no geometric information shall be used in order to refer to these vertices, some characterization at the topological level is necessary. The following representation by faces has shown to be useful in that regard [6]:

Definition: Let P be a coordination polyhedron. An *ordered face representation* of P can be obtained in the following way: Use elements of $\{1, ..., n\}$ to number the ligands $\{l_1, ..., l_n\}$ in a unique way. For every face f, the numbers of its vertices are arranged into a unique sequence as follows: P is viewed from outside and the vertices of f are collected clockwise starting with the smallest element.

Figure 1 shows a regular octahedron and an ordered face representation of it. For any polyhedron with the same topological view the same ordered face representation can be obtained by using an appropriate numbering for its vertices. For a polyhedron P with n vertices there are $n!$ different ways to number these vertices; the number of different ordered face representations, however, is $\frac{n!}{r}$. r is the order of the rotation group of that ideal polyhedron which has the same topological view as P and highest symmetry. This follows from the definition of an ordered face representation, which does not refer to the geometry of a polyhedron.

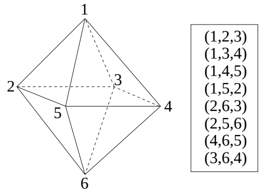


Figure 1: Ordered face representation.

Consider the topological view of a polyhedron P. An ordered face representation of P induces directions for the edges of the graph. For a digraph, rotations are automorphisms that do not change the direction of edges. As a consequence, a given ordered face representation of P is not changed as well by rotations. Consider the octahedron in Figure 1. Apply the rotation given by the permutation $(1563)(2)(4)$. For the resulting numbering of vertices we get the same ordered face representation of the octahedron.

The rotation groups of the Platonic solids are well known. For arbitrary polyhedra they can be determined using finding algorithms for geometric automorphism groups [14]. n-geometric automorphism groups of a graph can be displayed as symmetries of a drawing of the graph in n dimensions. Though the problem to determine whether a graph has a nontrivial geometric automorphism in two dimensions is NP-complete [15], it has been demonstrated that using a group-theoretic method is very efficient in practice for finding all 2- and 3-geometric automorphism groups of a graph [14]. Since the set of different topological views of coordination polyhedra in inorganic structures is finite, we can assume that the rotation group is given for every polyhedron under consideration.

2.2 Polyhedral clusters

Two coordination polyhedra of a structure can be connected by sharing common ligands. Three different forms of connection are of interest: vertex-, edge-, and face-sharing, which means that the polyhedra share one, two, or more common ligands, respectively. Connecting edges and faces must be edges and faces of both polyhedra and partial overlapping is not allowed. It is also possible that more than two polyhedra share a common vertex or edge.

Figure 2 shows face-sharing octahedra in sodium chloride and edge- as well as vertex-sharing octahedra and vertex-sharing tetrahedra in the silicate jadeite.

The ordered face representation allows to distinguish the

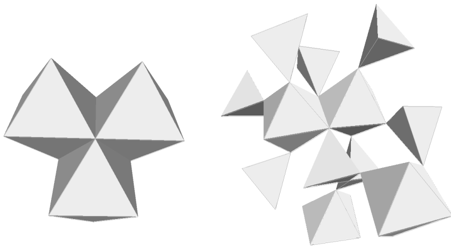


Figure 2: Coordination polyhedra in sodium chloride and the silicate jadeite.

relative orientation of polyhedra in clusters under certain conditions. Consider Figure 3 showing two pairs of square pyramids connected by an edge. There is no numbering scheme for the pyramids such that the ordered face representations and the labels of the connecting vertices become identical. In Figure 4 a) two tetrahedra of a chain of polyhedra are connected by a cube with connecting vertices belonging to the same face of the cube (a so-called trans-edge). In Figure 4 b) the connection is by vertices of different faces of the cube. There is no possibility to choose vertex numberings for the polyhedra such that the corresponding polyhedra of the chains have the same face oriented representation and the connecting vertices have the same number in both chains. This means that the face oriented representation allows to distinguish both chains.

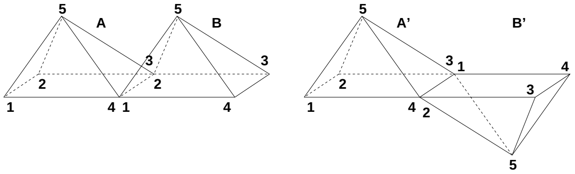


Figure 3: Non-equivalent pairs of pyramids.

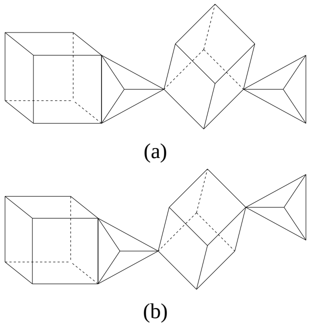


Figure 4: Non-equivalent chains of polyhedra.

Clusters of polyhedra can be described by graphs with nodes representing the polyhedra and edges representing their connections. The information added to the nodes and edges as labels depends on the usage of the graphs. In our approach we make use of topological as well as geometrical information on polyhedra. So we add an ordered face representation for each polyhedron together with the geometrical view as label to every node. The edges are labeled with one or more pairs of natural numbers identifying the polyhedra vertices involved in the corresponding connections. The following definition is from [6].

Definition: Let $\mathcal{P}$ be a cluster of polyhedra with an ordered face representation given for every polyhedron. A *polyhedra graph* for $\mathcal{P}$ is a graph $G_{\mathcal{P}} = (N_{\mathcal{P}}, E_{\mathcal{P}}, \lambda)$ with:

(1) $N_{\mathcal{P}}$ represents the polyhedra in $\mathcal{P}$; every node is labeled with the geometrical view and the ordered face representation of the corresponding polyhedron.

(2) $E_{\mathcal{P}} \subseteq N_{\mathcal{P}} \times N_{\mathcal{P}}$ is the set of directed edges representing every connection between polyhedra in $\mathcal{P}$ in both directions.

(3) $\lambda : E_{\mathcal{P}} \rightarrow 2^{\mathbb{N}\times\mathbb{N}}$ is a labeling function determining for every edge the pairs of vertex numbers involved in the connection between the polyhedra represented by the edge.

Obviously, with every edge a unique inverse is determined. For simplicity we do not remove this redundancy.

The ordered face representation of a polyhedron depends upon the chosen numbering scheme. Hence a polyhedra graph is unique up to these schemes. Figure 5 shows an example with edge- and vertex-sharing.

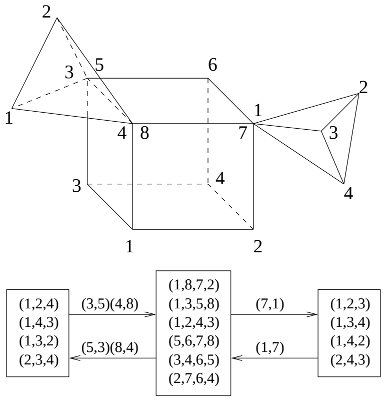


Figure 5: A cluster of polyhedra and its graph.

For labeled graphs to be isomorphic, all labels have to be preserved by the mapping. For topological equivalence the following definition is appropriate:

Definition: Let $G = (N_{\mathcal{P}}, E_{\mathcal{P}}, \lambda)$ and $G' = (N_{\mathcal{P}'}, E_{\mathcal{P}'}, \lambda')$ be two polyhedra graphs. Let $planes$ be a function which assigns the corresponding face representation to every polyhedron.

G and G' are *topologically isomorphic*, written $G \cong G'$, if the following holds:

Let S_i denote the symmetric group of degree i.

There are a bijection $\varphi : N_{\mathcal{P}} \longrightarrow N_{\mathcal{P}'}$ and a mapping $\pi : N_{\mathcal{P}} \longrightarrow \bigcup_{i=1}^{\infty} S_i, n \mapsto \pi_n$, such that

$$\begin{aligned}
&[\forall n \in N_{\mathcal{P}} : n, \varphi(n) \text{ are isomorphic}] \wedge \\
&[\forall n, n' \in N_{\mathcal{P}}\ \forall (i,j) \in \mathbb{N}^2 : \\
&(n, n') = e \in E_{\mathcal{P}} \wedge (i,j) \in \lambda(e) \\
&\Leftrightarrow \\
&(\varphi(n), \varphi(n')) = e' \in E'_{\mathcal{P}} \wedge (\pi_n(i), \pi_{n'}(j)) \in \lambda'(e')] \wedge \\
&[\forall n \in N_{\mathcal{P}} : f = (i_1, \ldots, i_m) \in \textit{planes}(n) \Leftrightarrow \\
&(f' = (\pi_n(i_k), \ldots, \pi_n(i_m), \pi_n(i_1), \ldots, \pi_n(i_{k-1})) \\
&\in \textit{planes}(\varphi(n)) \wedge \\
&\pi_n(i_k) = \min\{\pi_n(i_1), \ldots, \pi_n(i_m)\})]
\end{aligned}$$

This definition makes use of the ordered face representation of polyhedra and therefore takes the relative orientation of polyhedra into account.
Two clusters of polyhedra are *topologically equivalent* if their polyhedra graphs are topologically isomorphic.

The geometry of topologically equivalent clusters may differ strongly. Consider Figure 6 showing two rings of square pyramids sharing opposite edges of their square faces. In one of the rings the apices of the pyramids are directed to the outside and in the other ring they show to the inside. Since in both rings the number of pyramids is the same, they are topologically equivalent.

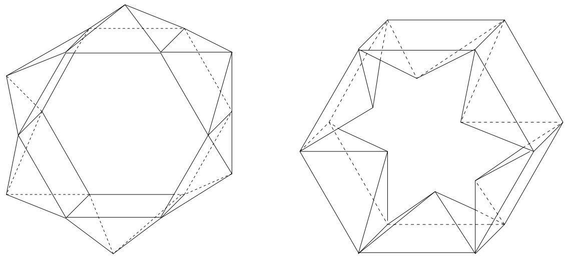


Figure 6: Topologically equivalent clusters.

Despite of this diversity of possible geometric realizations, the definition is practically useful since it does not exclude clusters from the set of candidates in the search for similar substructures when they differ with respect to the embedding in space or the form of their polyhedra but are identical with respect to the type of the polyhedra and their connections. Small deviations in the angles of connected polyhedra may sum up to quite large differences in the overall geometry, but it is difficult to provide a limit for differences being allowed. Furthermore, differences in the relative positioning of two neighbouring polyhedra in otherwise strongly similar clusters would lead to great differences when using methods like root mean square. Hence these differences are analysed in a second step and are not used for keeping structures out of the result.

An efficient method for determining topologically equivalent clusters of polyhedra in a given set of model structures is described in [6]. It uses a special index precomputed for the model structures in order to avoid the well-known complexity problems in connection with subgraph isomorphism. The method is applicable to crystal structures in general, i.e. to the search for finite clusters in infinite but periodic structures, and has been implemented as a web application [9]. In the following, we concentrate on the problem how to determine the geometric similarity of the target substructure and substructures of the infinite model structures; hence we only deal with finite clusters. For the modeling of infinite periodic crystals in general see [6].

3 The embedding problem

For the determination of equivalent substructures in model structures it is sufficient to find one mapping of the polyhedra of the search structure to polyhedra of the substructure such that the corresponding graphs are isomorphic. In order to check for geometric similarity, it is in general not sufficient to consider a single mapping for two graphs but all possible non-equivalent mappings have to be taken into account. The equivalence of mappings can result from symmetries in the model structures or from the existence of alternative permutations of the vertices of polyhedra.

3.1 Possible mappings

Consider the five-membered chain of tetrahedra shown in Figure 7. There are two ways to map the chain C considered as a cluster onto itself: mapping tetrahedron i to itself or - as indicated by C' - to tetrahedron $(4-i)'$, for $i = 0,..,4$. Obviously, the second mapping does not fit to the geometry of the chain indicated in the figure.
Consider the starlike cluster S of tetrahedra shown in Figure 7. It can be mapped to the cluster S' in three different ways. Only by using the mapping $i \rightarrow i', i = 0, ..., 3$, the cluster S can be moved exactly on S'.

The number of possible mappings can be exponential in the size of the given cluster. Consider the cycle in Figure 8 a). Let it represent a cycle of identical polyhedra with a single kind of connection. Assume a fixed embedding of the cycle into the graph shown in Figure 8 b) to be given. There exist seven further mappings which differ from the chosen mapping only with respect to the alternatives provided by the three 4-membered cycles. It is easy to see that in general for such a cycle with $5 \times n$ nodes there are 2^n possible mappings constructable in this way when the model structure has the same form as the structure in Figure 8 b) with

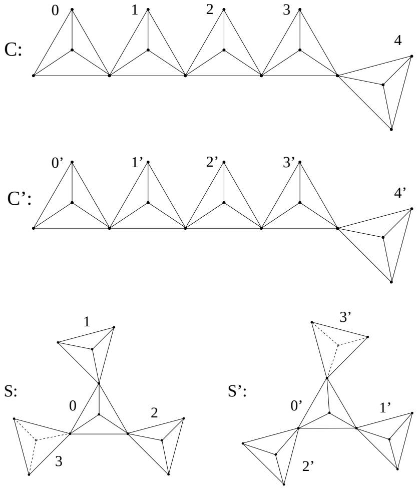


Figure 7: The embedding problem.

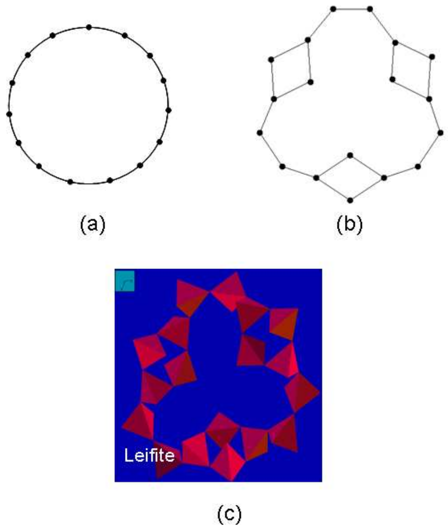


Figure 8: Exponential complexity.

$6 \times n$ nodes. Figure 8 c) demonstrates that this situation occurs in real crystal structures like the silicate leifite. There exist 315 non-translationally equivalent 15-membered rings of corner-sharing tetrahedra in this crystal (for a definition of rings in such crystal structures see [16]).

The number of mappings to consider can often be restricted by taking the symmetries of the target structure into account. It is sufficient to choose a single representative for each class of symmetrically equivalent substructures in the target structure. In leifite there are 44 different classes of 15-membered rings. Furthermore, when looking for embeddings given a representative of such a class only embeddings have to be considered which are not symmetrically equivalent. Consider again Figure 8. In the symmetry group (space-group) of leifite there is a threefold rotation axis in the center of the substructure shown in Figure 8 c). Let a concrete mapping of a 15-membered ring into the shown substructure be given. Then four additional mappings using the same tetrahedra of the substructure have to be considered instead of fourteen if no symmetry would be applicable.

3.2 Uniqueness criteria

To check two clusters for topological equivalence it is necessary to find for every polyhedron in one of the corresponding polyhedra graphs a suitable permutation of the vertices such that both graphs become identical (up to the geometric information). Though the number of permutations for a polyhedron to consider can be restricted by taking the rotation group into account, the resulting set can still be quite large. Therefore, all possibilities to restrict this set further should be applied. The following properties of a convex polyhedron P and an ordered face representation of it are helpful into that regard:

- Every edge of P belongs to exactly two faces.
- Every face of P has at least three edges.
- The representation of faces implies that every edge shows up as a pair (i, j) in one face and as (j, i) in the other face.

We get as an immediate consequence: Let P and P' be two polyhedra in ordered face representation. Let ϕ be an isomorphism between P and P' and $e = (i, j)$ an edge of P. Then there is exactly one permutation π of the vertices of P' such that the following holds: $(\pi(\phi(i)), \pi(\phi(j))) = (i, j)$ and P and P' have the same ordered face representation.

The uniqueness of the permutation π follows from the following observation: e and $\phi(e)$ shall have the same numbers; hence $\pi(\phi(i)) := i$ and $\pi(\phi(j)) := j$. There are exactly two faces F_1 and F_2 of P which are incident with e. Without loss of generality it can be assumed that e occurs in the representation of F_1 in the order i, j and in the representation of F_2 in the order j, i. For every sequence i, j there is exactly one representation of a face in the ordered faced representation of a polyhedron. This means that the number of every vertex h in $\phi(F_1)$ has to be mapped by π to the number of $\phi^{-1}(h)$ in order to get the same sequence of vertex numbers as it occurs in the representation of F_1. The same holds for $\phi(F_2)$. So π is uniquely determined for the vertices of $\phi(F_1)$ and $\phi(F_2)$. Since there are now further edges of P' determined, the permutation is uniquely fixed for all vertices of P'. The vertices of an edge being involved in a connection of a cluster of polyhedra must be numbered identically to the corresponding edge in a topologically equivalent cluster. Hence in case of edge- or face-sharing the ordered face representation guarantees that the vertex numbering in one of the corresponding polyhedra graphs determines uniquely the vertex numbering in the other graph. A similar situation is given when two polyhedra are linked by the two vertices of an edge e of another polyhedron P. The permutation of the polyhedron P' in the isomorphic graph is determined uniquely since the vertices of e and the corresponding edge in the isomorphic graph have to be identical.

Example: Consider the two chains of tetrahedra in Figure 9 a). They are topologically equivalent. A permutation π of P' of a topological isomorphism for the two chains must map 2 to 3 and 4 to 1. The edge $e = (3, 1)$ of P taken in the given order determines the face $(1, 2, 3)$ of P. The face in P' with $(2, 4)$ occuring in this order in its representation is $(2, 4, 1)$. We therefore get $\pi(1) = 2$ and $\pi(3) = 4$.

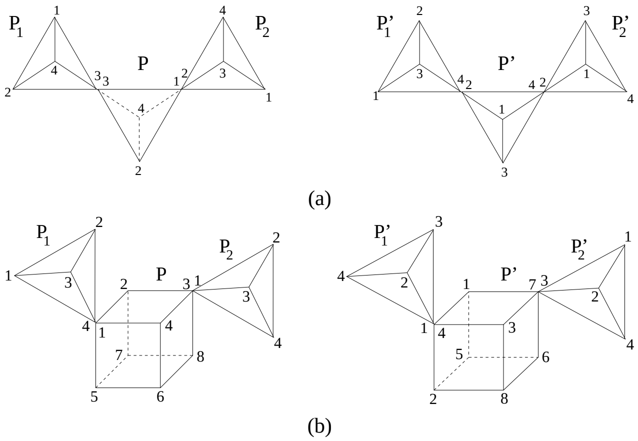


Figure 9: Connections fixing a permutation.

For the uniqueness of the permutation of a polyhedron P' it is even sufficient when the vertices of P involved in the connection are vertices of the same face F: The vertices of the corresponding face F' in P' are uniquely determined by the representation of F since no pair of vertices of a face which are not the endpoints of an edge can both be vertices of a further face. Otherwise, the second face would intersect F. Hence there is a single entry with both vertices in the representation of P. For an example, see Figure 9 b). Vertices 1 and 3 of P occur together only in face $(1, 2, 3, 4)$. From the connections follows that the vertices 4 and 7 of P' have to become 1 and 3, respectively. This implies the change of 1 to 2 and of 3 to 4.

In case the permutation of a polyhedron cannot be fixed by its neighbourhood, a representative of each class of permutations resulting in the same ordered face representation has to be considered. Take a single polyhedron, an octahedron, for example, in its regular form. There are $6!$ possible ways to label its vertices with different numbers. There are, however, much less ordered face representations of an octahedron, namely 30. This results from the properties of its symmetry group $m\bar{3}m$. It contains three perpendicular fourfold rotations which are automorphisms.

Coordination polyhedra may have distortions which can be used to restrict the set of geometrically reasonable permutations when only a single vertex is fixed by a connection. This is a consequence of missing symmetries (the topological view of polyhedra could be augmented by symmetry information such that it is not only sufficient to have identical face representation but the symmetry groups must fulfill conditions as well).

For a given cluster there can exist many isomorphic subgraphs in the same model graph. Hence the problem arises which ones to present in the result of a query. Clusters with isomorphic polyhedra graphs can have quite different geometries. We need a method to rank results which is fast to compute and which nonetheless is suited to measure the differences in the geometries of clusters since the number of qualifying model structures can be very large as well.

4 Geometric similarity of embeddings

Geometric similarity of three-dimensional structures has been defined in various different forms depending on the underlying applications. For the comparison of clusters of polyhedra we need a definition of similarity which takes into account that coordination polyhedra represent strong bonds and that connections between polyhedra are given by the sharing of common atoms.

4.1 The problem

For molecular structures, operations such as rotations, translations, and scaling as well as the editing of molecules including the deletion or insertion of atoms are sometimes applied to check for a possible matching [17]. Other approaches look for maximal subgraphs allowing distances of atoms to vary in fixed ranges [5] or identify backbone structures in order to find rigid motions for solving the matching problem for substructures [18].

For inorganic structures “the use of the term ’similar’, in the definition of configurational and crystal chemical isotypism, arises from the inherent difficulty in defining a priori limits on the similarity of geometrical configurations or physical/chemical characteristics” [19]. This means that flexibility has to be taken into account like for organic structures.

Since for inorganic structures the notion of coordination polyhedra is fundamental [8], it should be possible to define and analyze similarity at the level of polyhedral networks. A single coordination polyhedron characterizes the bonding structure for a distinguished central atom and neighbouring atoms. Mostly, these local bonds are strong compared to other bonds. The form of a polyhedron depends upon the kind of the central atom, the number and kind of its ligands, and other atoms in the vicinity. In a typical silicate, for example, four oxygen atoms with strong bonds to a silicon atom form a regular tetrahedron. Sodium and chlorine atoms are forming regular octahedra in sodium chloride whereas the octahedra formed by lithium and oxygen atoms in some silicates are quite irregular.

Distortions of polyhedra from the perfect form can be measured [13], [20], [21], but it is also argued that no unique index can be defined for measuring the size of the distortion since measures are never completely model-free [22], [13]. Therefore, when comparing substructures of inorganic compounds at the level of coordination polyhedra, a distinction between the placement of polyhedra in the structure and the shape of polyhedra seems to be appropriate. The placement of a polyhedron can be described by the coordinates of its central atom. A difference in the coordinates of the central atoms of two corresponding polyhedra implies that the coordinates of the ligands are different as well or that there are differences in the shape of the two polyhedra.

The reference coordinate axes of two structures cannot be assumed to be identical. Hence rotations and translations are normally necessary to check for geometric isomorphism. The most restrictive definition of geometric similarity requires a bijection to exist between the two given point sets such that the coordinates of a point and its image in the other set are allowed to differ only in the range of a given small tolerance. This tolerance is necessary in connection with experimental data. A weakened form of this definition assumes polyhedra to be rigid bodies according to the fact that they model strong bonds. It considers vertex- and edge-sharing connections of polyhedra in clusters as 'ball-and-sockets' and 'hinges'. Whereas a face-sharing allows no flexibility, a vertex-sharing and an edge-sharing of two polyhedra allow the central atoms to move on the surface of a sphere and on circles, respectively, depending on the overall flexibility of the cluster. For two clusters it should be possible to apply appropriate motions such that the resulting clusters are geometrically similar according to the definition above. A distinction can be made whether motions with intermediate interpenetrating polyhedra are allowed or not.

Figure 10 shows an example where two structures become identical if an appropriate hinge motion is applied. It should be mentioned that these two rings of tetrahedra are normally considered as strictly distinct clusters.

4.2 Polyhedral clusters as joint structures

For the mathematical modeling of geometric embeddings of polyhedra graphs the algebra presented in [23] is well suited. It uses projective geometry as underlying theory and allows to deal elegantly with motions and geometric transformations. We first repeat some definitions of [23] (for an introduction into projective geometry see [24], for example).

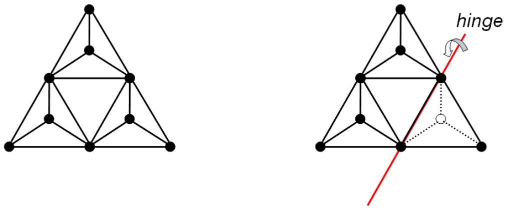


Figure 10: Identification by hinge motion.

Since we are only interested in embeddings into the Euclidean space it is sufficient to consider projective spaces over the real numbers.

Let $n \in \mathbb{N}$. For $x, y \in \mathbb{R}^{n+1} \setminus \{\mathbf{0}\}$ define the equivalence relation

$$x \sim y \qquad \text{:iff} \qquad \text{there is a } \lambda \in \mathbb{R} \text{ with } x = \lambda y.$$

The set of equivalence classes

$$\mathbb{PR}^n := (\mathbb{R}^{n+1} \setminus \{\mathbf{0}\}) / \sim$$

is called *the projective space of dimension* n *over* $\mathbb{R}$; the elements of $\mathbb{PR}^n$ are called *projective points.*
Let $x \in \mathbb{PR}^n$ with $x = (x_0, \ldots, x_n)$. Then x and λx describe the same projective point for all $\lambda \neq 0$. The equivalence class of x is usually denoted as $(x_0 : \cdots : x_n)$ and $(x_0 : \cdots : x_n)$ are called *homogeneous coordinates* of the point x.
For $x \in \mathbb{R}^3$ with $x = (x_1, x_2, x_3)$ it is convenient to use the point

$$\bar{x} := (x, 1) := (x_1, x_2, x_3, 1) \in \mathbb{PR}^3$$

as representative of its equivalence class.
If $\varphi : M \longrightarrow \mathbb{R}^3$ is a mapping of an arbitrary set M into $\mathbb{R}^3$, then $\bar{\varphi}$ shall denote the continuation of φ on the homegeneous coordinates of the images, i.e.

$$\bar{\varphi}(m) := \overline{\varphi(m)}$$

for all $m \in M$.
For the description of all lines of $\mathbb{R}^3$ properties and notions of the Grassman-Cayley algebra are used. A general multiplication on the points of $\mathbb{PR}^n$ is defined as follows:
Let $a, b \in \mathbb{PR}^3$ with $a = (a_1, a_2, a_3, a_4)$ and $b = (b_1, b_2, b_3, b_4)$. Consider the matrix

$$D_{a,b} := \begin{pmatrix} a_1 & a_2 & a_3 & a_4 \\ b_1 & b_2 & b_3 & b_4 \end{pmatrix}.$$

The *outer product* is defined as

$$\vee : \mathbb{PR}^3 \times \mathbb{PR}^3 \longrightarrow \mathbb{R}^6,$$

$$a \vee b := (d_{14}, d_{24}, d_{34}, d_{23}, d_{31}, d_{12}),$$

where d_{ij} denotes the 2×2-minor obtained from the ith and jth column of $D_{a,b}$.
The 6-tupel $a \vee b$ is called 2-*extensor* of the points a and b; it is also written in the form ab.
The set of all such 6-tupels for pairs of points is denoted by $\mathbf{L}^2$.
Let $a, b \in \mathbb{R}^3$ and $\bar{a}$, $\bar{b}$ be the equivalent projective points. Then the following holds:
$\bar{a}\bar{b} = \bar{a} \vee \bar{b} = (a - b, a \times b)$.
$(a - b, a \times b)$ are called the *Plücker coordinates* of a and b. Not all elements of $\mathbb{R}^6$ correspond to pairs of points but every such 6-tuple represents a *screw* in $\mathbb{R}^3$. This results from the fact that every element of $\mathbb{R}^6$ can be obtained as sum of two 2-extensors. The set of all 6-tuples or screws is usually denoted by Λ^2.

Consider the polyhedra graph $G_{\mathcal{P}} = (N_{\mathcal{P}}, E_{\mathcal{P}}, \lambda)$ for a polyhedral cluster $\mathcal{P}$ together with the geometrical views of the polyhedra. The edges $E_{\mathcal{P}}$ of the graph can be partitioned according to their corresponding type (vertex, edge, or face):
$E_{\mathcal{P}} = E_v \dot{\cup} E_e \dot{\cup} E_f$.
Define the following mapping

$$\mathbf{H} : E_e \longrightarrow \mathbf{L}^2,$$

$$e = (P_i, P_j) \mapsto \overline{pos(L_{e_1})} \vee \overline{pos(L_{e_2})} =: L_{i,j},$$

where $\{L_{e_1}, L_{e_2}\} = V_{P_i} \cap V_{P_j}$ for every edge $(P_i, P_j) \in E_e$; V_{P_i}, V_{P_j} denote the sets of vertices of the polyhedra $P_i, P_j \in \mathcal{P}$, respectively.
Since for every edge $(P_i, P_j) \in E_e$ we have

$$L_{i,j} = -L_{j,i},$$

it follows that $\mathbf{H}$ is a hinge motion [25].

We are now ready to define when a polyhedral cluster is flexible and when two given polyhedra graphs can be considered as geometrically transformable into each other by motions which respect the connections of the polyhedra.

Definition: Let $G = (N_{\mathcal{P}}, E_{\mathcal{P}}, \lambda)$ be a polyhedra graph with geometrical views given for all polyhedra. Let

$$\mathbf{S} : N_{\mathcal{P}} \longrightarrow \Lambda^2, P_i \mapsto \mathbf{S}_i$$

be a mapping such that the following holds:

- for every connection by vertices $(P_i, P_j) \in E_v$ with $V_{P_i} \cap V_{P_j} = \{L\}$ we have

$$\mathbf{S}_i pos(L) = \mathbf{S}_j pos(L),$$

- for every connection by edges $(P_i, P_j) \in E_e$ there exists a scalar $\lambda_{i,j} \in \mathbb{R}$, such that

$$\mathbf{S}_i - \mathbf{S}_j = \lambda_{i,j} L_{i,j},$$

- for every connection by faces $(P_i, P_j) \in E_f$ we have

$$\mathbf{S}_i = \mathbf{S}_j.$$

$\mathbf{S}$ is a *joint motion* of the configuration induced by pos. $\mathbf{S}$ is called *trivial* if $\mathbf{S} \equiv D$ for some $D \in \Lambda^2$.
G is called *flexible* if it admits a non-trivial joint motion $\mathbf{S}$.

Since the geometrical views of the polyhedra of a polyhedra graph determine a configuration of the graph, it is possible to define the congruency of two polyhedra graphs in analogy to general graphs. Based on this definition, conditions for the possibility to transform the graphs into each other by motions can be given.

Definition: Let $G_1 = (N_1, E_1, \lambda_1)$ and $G_2 = (N_2, E_2, \lambda_2)$ be two polyhedra graphs with functions pos_1 and pos_2 assigning the coordinates to the vertices of the polyhedra of G_1 and G_2, respectively.
G_1 und G_2 are *congruent* if there exists a topological isomorphism $\varphi : G_1 \longrightarrow G_2$ and a $\mathbb{R}^3$-isometry T such that

$$T \circ pos_1 = pos_2 \circ \varphi.$$

Let $SE(3)$ denote the set of proper rigid motions of $\mathbb{R}^3$. Let $\mathbf{M} : \Lambda^2 \longrightarrow SE(3)$ be a mapping assigning the homogeneous representation to every screw. If a motion $\mathbf{S}$ of G_1 exists such that G_1 after applying the (homogeneous representation) of the motion $\mathbf{S}$ and G_2 are congruent with respect to pos_1 and pos_2, i.e., if there exists an $\mathbb{R}^4$-isometry T such that for all $n_i \in N_1$

$$(T \circ \mathbf{M}(\mathbf{S}_i) \circ \overline{pos_1})(n_i) = (\overline{pos_2} \circ \varphi)(n_i)$$

holds, then $\mathbf{S}$ is called a *geometric transformation* of G_1 into G_2 and G_1 is called *geometrically transformable into G_2 by joint motions.*

This definition fixes when two polyhedra graphs can be transformed into each other geometrically. The underlying assumptions are that polyhedra are rigid bodies and that only joint motions can be applied.

Consider a set of topologically isomorphic polyhedra graphs. Assume that corresponding polyhedra in the graphs are congruent. The question arises whether for any pair of graphs (G_1, G_2) in the set, G_1 is geometrically transformable into G_2 by joint motions. Look again at the rings of polyhedra in Figure 6. Applying hinge motions and moving the pyramids through one another allows to transform one ring into the other. However, if we add two handles to each of the rings with edge-sharing pyramids and with connections to the ring at opposite pyramids again by sharing edges (a kind of 'crown' is built), the resulting clusters are topologically equivalent but no geometric transformation is possible. The reason for the missing of a transformation is that the pyramids of the handles as well as those of the ring have to be moved through one another in order to change the direction of their apices. Applying such a motion to one of the handles results in a squeezing of the other handle since each handle and the ring can only be moved perpendicular to their hinges. This follows from the fact that the complete structure has only connections by edges.

Two non-flexible topologically isomorphic clusters which are not geometrically transformable into one another can be constructed as follows: Take a cube and place a pyramid on each of its faces; remove the cube from the cluster. The resulting configuration of edge-sharing pyramids is topologically equivalent to the cluster which we obtain when the apices of the pyramides are all pushed in. Since the two clusters are not flexible, no geometric transformation is possible.

The notion of geometric transformation can be weakened in order to get a definition of similarity which takes into account that in crystal structures we do not deal with ideal polyhedra. One idea is to restrict the pos-functions in the definition of a geometric transformation to vertices involved in connections and to allow differences in the coordinates of corresponding vertices up to some fixed limit. A second possibility is to restrict these functions to the central atoms of the polyhedra with the same relaxation with respect to the coincidence of coordinates.

When two structures are geometrically similar to one another according to one of these definitions they can be further analysed for the similarity of corresponding polyhedra. Symmetries can be checked, for example, the kinds of atoms can be compared, or measures of distortions can be used. In the following, we concentrate on the first step. Furthermore, we do not consider non-trivial joint motions to change the geometry of structures. We rather use the root mean square method to measure the 'distance' of two structures with respect to the positions of the central atoms of their polyhedra.

5 Implementation and results

The problem of minimization of distances for two sets of points in three-dimensional space is sometimes called 'problem of absolute orientation'. Two well-known methods for solving this problem have been investigated: the algorithm of B.K.P. Horn [26], a closed-form solution of the problem using unit quaternions to represent rotations, and the algorithm of W.A. Dollase [27] using infinitesimal rotations. We have given preference to the algorithm of Horn since it avoids the potentiation of numeric instabilities, which may arise in connection with orthonormalization in the algorithm of Dollase.

Let $G_{\mathcal{P}} = (N_{\mathcal{P}}, E_{\mathcal{P}}, \lambda_{\mathcal{P}})$ and $G_{\mathcal{P}'} = (N_{\mathcal{P}'}, E_{\mathcal{P}'}, \lambda_{\mathcal{P}'})$ be two finite polyhedra graphs with $N_{\mathcal{P}} = \{P_1, \ldots, P_n\}$ and $N_{\mathcal{P}'} = \{P'_1, \ldots, P'_n\}$.

Let $\varphi : \mathcal{P} \longrightarrow \mathcal{P}', P_i \mapsto P'_i$, be an isomorphism and let $C_{\mathcal{P}} = \{c_1, \ldots, c_n\}$ and $C_{\mathcal{P}'} = \{c'_1, \ldots, c'_n\}$ be the sets of coordinates of the central atoms of $\mathcal{P}$ and $\mathcal{P}'$, respectively (referring to a common Cartesian coordinate system), i.e. $c_i = pos_{\mathcal{P}}(P_i)$ and $c'_i = pos_{\mathcal{P}'}(P'_i)$, for $i = 1, \ldots, n$. Here pos is the function from polyhedra to the coordinates of their central atoms derived from the geometrical views of the polyhedra.

φ induces a mapping

$$\psi : C_{\mathcal{P}} \longrightarrow C_{\mathcal{P}'},$$

$$pos_{\mathcal{P}}(c_P) = c_i \mapsto c'_i = pos_{\mathcal{P}'}(c_{\varphi(P)})$$

that sends the coordinates of the central atom of every polyhedron of $\mathcal{P}$ to the coordinates of the central atom of its image under φ.

The sets $C_{\mathcal{P}}$ and $C_{\mathcal{P}'}$ are now considered as rigid subsets of $\mathbb{R}^3$, which shall be moved such that the best possible matching results. This means that a motion $T \in SE(3)$ is looked for solving the following least-squares problem:

$$U := \sum_{P \in N_{\mathcal{P}}} ||(pos_{\mathcal{P}'} \circ \varphi)(P) - (T \circ pos_{\mathcal{P}})(P)||_2^2 = \min .$$

With the notations from above we get the following equivalent formulation:

$$U := \sum_{i=1}^{n} ||c'_i - T(c_i)||_2^2 = \min .$$

An optimal rigid Euclidean motion T moves the centroid of $C_{\mathcal{P}}$ to the centroid of $C_{\mathcal{P}'}$ [26]. By placing centroids into the origin of the common coordinate system, the least-squares problem reduces to the determination of a rotation T solving the equation above for $C_{\mathcal{P}}$ and $C_{\mathcal{P}'}$ relative to their centroids.

For measuring the similarity of two structures $\mathcal{P}$ and $\mathcal{P}'$ the root mean square

$$\epsilon := \frac{\sqrt{U}}{n}$$

is taken. The similarity increases with decreasing ϵ. What we have to keep in mind is that polyhedra distortions and rotations allowed by their connections may lead to differences in the shape of otherwise similar clusters.

For our problem it is sufficient to consider quaternions as unit vectors of $\mathbb{R}^4$. Each unit quaternion uniquely represents a rotation in three-dimensional space. Furthermore, scaling of point sets is not wanted since motions shall be rigid. Therefore, we assume the scaling factor in the algorithm of Horn to be 1.

Algorithm:

Input: Two finite topologically isomorphic clusters of polyhedra C_1 and C_2 with sets of polyhedra $\{P_1, ..., P_n\}$ and

```
$\{P'_1, ..., P'_n\}$, respectively, and a topological isomorphism
$\varphi$ such that $P'_i = \varphi(P_i)$, for $i = 1, ..., n$.
Output: Root mean square of the deviation of $C_1$ and $C_2$
after an optimal distance minimizing movement of the central atoms according to $\varphi$ in a common coordinate system.
// Initialization
Transform the coordinates of all atoms of $C_1$ and $C_2$ into
Cartesian coordinates; generate the sets $A_{C_1} = \{c_1, ..., c_n\}$
and $A_{C_2} = \{c'_1, ..., c'_n\}$ of coordinates of the central atoms
of $C_1$ and $C_2$, resp.
Centralize $A_{C_1}$ and $A_{C_2}$:
    $\bar{c} := \frac{1}{n} \sum_{i=1}^{n} c_i$;
    $\bar{c'} := \frac{1}{n} \sum_{i=1}^{n} c'_i$;
    for $i = 1, ..., n$ do
        $c_i := c_i - \bar{c}$;
        $c'_i := c'_i - \bar{c'}$;
    end
// Solution of the balancing problem
Compute with the algorithm of Horn the rotation matrix $R_q$,
which moves $A_{C_1}$ optimally on $A_{C_2}$.
// Computation of the result and parameter
// extraction (rotation angle $\Theta_{res}$ and axis
// $(l_{res}, m_{res}, n_{res}) \in \mathbb{R}^3$)
    $U := \sum_{i=1}^{n} \|c'_i - R_q(c_i)\|_2^2$;
    $\epsilon := \frac{\sqrt{U}}{n}$;
    Compute $\Theta_{res}, l_{res}, m_{res}, n_{res}$ given $R_{res} = R_q$.
```

The implementation of the algorithm in C++ has been integrated into our system POLYSEARCH [6]. A graphical interface of POLYSEARCH allows to mark a substructure of a chosen crystal for search. Topologically equivalent substructures in a set of model structures are determined based upon a representation of the crystals by periodic graphs. When building the result, the number of substructures of the same crystal structure being geometrically equivalent to each other is reduced by exploiting the symmetries of the structure. The information on the subgraphs corresponding to the remaining substructures is used to determine the ranking, which can be done fast (cp. [28]). The execution time of the search for equivalent substructures mainly depends on the selectivity of the search structure relative to the given set of model structures [6]. The extension of our web application [9] by an appropriate graphical interface for the representation of the geometric similarity of substructures is in progress.

Figure 11 shows screenshots of a search structure in the silicate aminoffite and three substructures of the result with their root mean square values (RMS). In this example, the structures consist of tetrahedra with nearly identical geometry. The different RMS values mainly result from the differences in the orientation of the tetrahedra in the clusters.

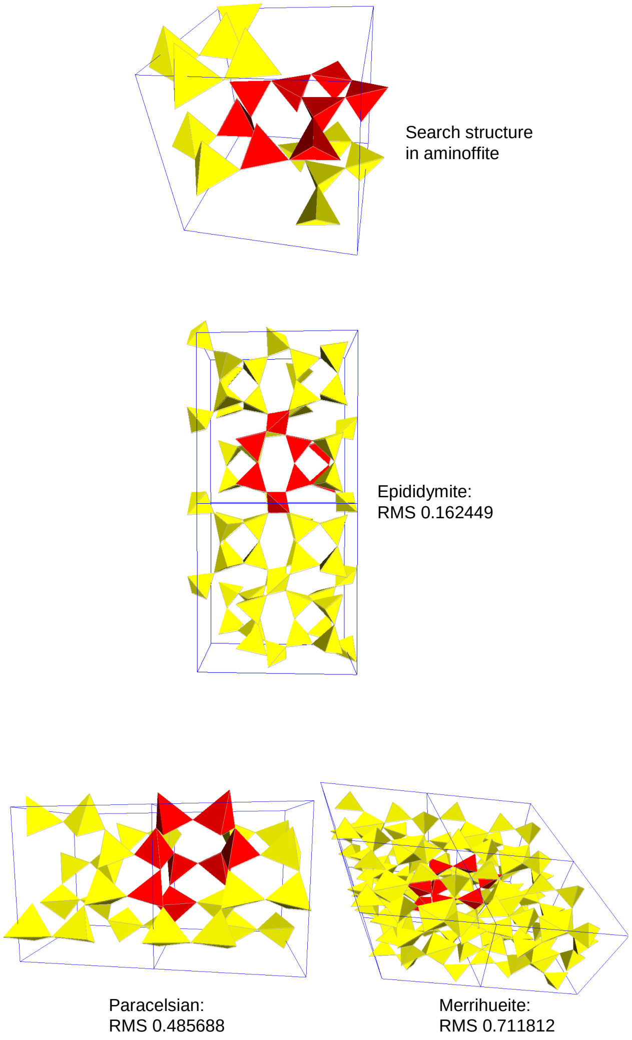


Figure 11: Search structure and three similar substructures.

6 Conclusions

In this paper, the main focus has been to show how geometric similarity of clusters of coordination polyhedra can be defined and computed by an appropriate algorithm. The integration of the algorithm into our system for the retrieval of isomorphic substructures in large sets of model structures allows to search for geometrically similar substructures in large inorganic crystal structure databases. The search can be combined with other information about the target structure such as publication data, symmetries, or the assignment to a special class.

A further application can be seen in the field of structure prediction. In recent years, the enumeration of hypothetical inorganic structures has attracted much attention [29], [30].

Geometric embeddings are computed for graphs which are generated in an enumeration process. It is very helpful to analyse these hypothetical structures with respect to their similarity with real structures.

As mentioned above, our system only checks for the similarity of point sets built from the coordinates of central atoms. In the future, we intend to implement methods for taking all information about polyhedra into account and to provide information about motions, which can be applied to get a better matching of two topologically isomorphic structures. A further idea is to apply methods developed in the field of maximal subgraph search in order to be able to compare structures without referring to a target substructure.

Recently, a graphical interface has been implemented in POLYSEARCH offering operations for constructing polyhedral clusters artificially. A collection of different kinds of polyhedra as often found in crystal structures is available and methods for constructing clusters and modifying them by applying joint motions have been realized. For the generated clusters, embeddings into polyhedral networks of real crystal structures can be determined and ranked.

References

[1] H.-J. Klein and Ch. Mennerich. *Searching and ranking similar clusters of polyhedra in crystal structures*, Proc. 2nd Int. Conf. ADVCOMP, Valencia, Spain, IEEE, pp. 235-240, 2008

[2] F.H. Allen. *The Cambridge Structural Database: a quarter of a million crystal structures and rising*, Acta Cryst. B58, pp. 380-388, 2002

[3] A. Belski, M. Hellenbrandt, V.L. Karen, and P. Liksch. *New developments in the Inorganic Crystal Structure Database (ICSD): accessibility in support of materials research and design*, Acta Cryst. B58, pp. 364-369, 2002

[4] J.A. Chisholm and S. Motherwell. *A new algorithm for performing three-dimensional searches of the Cambridge Structural Database*, J. of Appl. Cryst. 37, pp. 331-334, 2004

[5] J.W. Raymond and P. Willett. *Similarity searching in databases of flexible 3D structures using smoothed bounded distance matrices*, J. Chem. Inf. Comput. Sci. 43, pp. 908-916, 2003

[6] H.-J. Klein. *Retrieval of isomorphic substructures in crystallographic databases*, Proc. 16th Int. Conf. SSDBM, Santorini, Greece, IEEE, pp. 255-264, 2004

[7] M. Kuramochi and G. Karypis. *Discovering frequent geometric subgraphs*, Inform. Systems 32, pp. 1101-1120, 2007

[8] M. O'Keeffe and B.G. Hyde. *Crystal Structures: 1. Patterns and Symmetry*, Mineralogical Society of America, 1996

[9] http://www.is.informatik.uni-kiel.de/∼hjk/crystana

[10] D.R. Chand and S.S. Kapur. *An algorithm for convex polytopes*, J. of the ACM 17(1), pp. 78-86, 1970

[11] E. Prince (Ed.). *International Tables for Crystallography - Vol. C*, Part 9, online edition, Springer, 2006

[12] P. Villars and K. Cenzual. *Pearson's Crystal Data: Crystal Structure Database for Inorganic Compounds*, Release 2008/9, ASM International, 2008

[13] E. Makovicky and T. Balić-Zunić. *New measure of distortion for coordination polyhedra*, Acta Cryst. B54, pp. 766-773, 1998

[14] D. Abelson, S.-H. Hong, and D.E. Taylor. *Geometric automorphism groups of graphs*, Discr. Appl. Mathematics 155, pp. 2211-2226, 2007

[15] A. Lubiw. *Some NP-complete problems similar to graph isomorphism*, SIAM J. Comput. 10(1), pp. 11-21, 1981

[16] K. Goetzke and H.-J. Klein. *Properties and efficient algorithmic determination of rings in finite and infinite polyhedral networks*, J. of Non-Crystalline Solids 127, pp. 215-220, 1991

[17] X. Wang and J.T.L. Wang. *Fast similarity search in three-dimensional structure databases*, J. Chem. Inf. Comput. Sci. 40, pp. 442-451, 2000

[18] L.P. Chew, D. Huttenlocher, K. Kedem, and J. Kleinberg. *Fast detection of common geometric substructure in proteins*, Proc. 3rd ACM RECOMB, pp. 104-112, 1999

[19] J. Lima-de-Faria et al. *Nomenclature of inorganic structure types*, Acta Cryst. A46, pp. 1-11, 1990

[20] T.B. Balić-Zunić and E. Makovicky. *Determination of the centroid or 'the best centre' of a coordination polyhedron*, Acta Cryst. B52, pp. 78-81, 1996

[21] E. Lalik. *Shannon information as a measure of distortion in coordination polyhedra*, Appl. Cryst. 38, pp. 152-157, 2005

[22] I.D. Brown. *On measuring the size of distortions in coordination polyhedra*, Acta Cryst. B62, pp. 692-694, 2006

[23] W. Whiteley. *Rigidity of molecular structures: generic and geometric analysis*, in: P. Duxbury and M. Thorpe (eds.), *Rigidity and Applications*, Kluwer, 1999

[24] H.S.M. Coxeter. *Projective Geometry*, 2nd ed., Springer, 2003

[25] H. Crapo and W. Whiteley. *Statics of frameworks and motions of panel structures, a perspective geometric introduction*, Structural Topology 6, pp. 43-82, 1982

[26] B.K.P. Horn. *Closed-form solution of absolute orientation using unit quaternions*, J. Optical Soc. of America A4, pp. 629-642, 1987

[27] W.A. Dollase. *A method on determining the distortion of coordination polyhedra*, Acta Cryst. A30, pp. 513-517, 1974

[28] A. Lorusso, D.W. Eggert, and R.B. Fisher. *A comparison of four algorithms for estimating 3-D rigid transformations*, Proc. Brit. Conf. on Machine Vision, pp. 237-246, 1995

[29] H.-J. Klein. *Systematic generation of models for crystal structures*, Proc. 10th Int. Conf on Mathem. and Comp. Modelling and Scientific Comp., Boston, 1995, in: Mathematical Modelling and Scientific Computing 6, 325-330, 1996

[30] M.D. Foster, M.M.J. Treacy. *A database of hypothetical zeolite structures*, http://www.hypotheticalzeolites.net

Optimising the Quality of Experience during Channel Zapping

The Impact of Advertisements during Channel Zapping

R.E. Kooij[1,2], V.B. Klos[1], B.E. Godana[2], F.P. Nicolai[2]. O.K. Ahmed[1]
[1]TNO Information and Communication Technology
Delft, the Netherlands
{robert.kooij,victor.klos,kamal.ahmed}tno.nl
[2]Fac. of Electrical Engineering, Mathematics and Computer Science
Delft University of Technology
Delft, the Netherlands

Abstract—**Nowadays various digital television services are available. However, the user of these services experiences longer delays than the traditional analog TV while switching from channel to channel. The digital TV operator usually displays a black screen with the channel number during zapping. However, it could be interesting for the TV viewer, if the operator displays a screen with information instead of just a black screen. This information may be an advertisement, information about the target channel, personalized content of the user etc. In this paper, we describe a subjective experiment where the Quality of Experience (QoE) of channel zapping was quantified, while displaying a random set of advertisement pictures during zapping. It is found that, for longer zapping times, advertisements give better QoE than the black screen. However, when zapping times are small, users prefer a black screen over a glance of an advertisement picture. Based upon our findings we propose a system for optimal zapping experience. The system first estimates the zapping time (per target channel) and then, depending on this estimation, displays either a black screen, a picture or a clip.**

Keywordst; Channel Zapping, Quality of Experience, Mean Opinion Score, IPTV, advertisements.

I. Introduction

This paper is an extended version of [1]. The extensions and new contributions over [1] are as follows: the number of persons that have participated in the subjective experiments has been increased from 12 to 30; a subsection has been added about measurements that have been performed to get insight about zapping times for today's digital television services; a completely new section has been added about how the finding of this research can be used to design a system for optimal zapping experience.

Telecom Service Providers around the world are in a race to deploy new revenue generating services in order to offset the accelerating decline in voice revenues. For instance, US based providers faced a decline of 34% in voice-related revenues between 2000 and 2006 [2]. Among others, Service Providers came up with a new service called "triple play", which is the commercial bundling of voice, video and data on a common IP based network infrastructure. This IP based network infrastructure allows providing enhanced applications and services such as IPTV, VoIP, video telephony and Video on demand (VoD). However, as providers deploy new services, they also have to provide optimal Quality of Experience (QoE). QoE takes into account how well a service meets customers goals and expectations rather than focusing only on the network performance. In this highly competitive market Service Providers which are offering high quality IPTV services should address the QoE requirements of IPTV.

One of the key elements of QoE of IPTV is how quickly users can change between TV channels, which is called channel zapping. The zapping time is the total duration from the time that a viewer presses the channel change button, to the point the picture of the new channel is displayed, along with the corresponding audio. Minimum quality requirements for a lot of aspects related to IPTV have been specified by both the ITU [3] and the DSL Forum [4]. However in the ITU document there are no recommendations at all related to zapping times, while in the DSL forum document it is recommended to limit zapping time to an arbitrary maximum of 2 seconds. Additionally it is noticed in the document that providers should strive for zapping times in the order of 1 second.

Because these quality requirements are rather vague Kooij et al. [5] conducted a number of subjective tests in order to get insight in the relation between QoE and zapping time. For the tests described in [5], during channel zapping a black screen was visible which contained the number of the target channel. The QoE was expressed as a so-called Mean Opinion Score (MOS). The test subjects (21 in total) could select one of the following five opinion scores, motivated by the ITU-T ACR (Absolute Category Rating) scale, see [6]: 5: *Excellent zapping quality*, 4: *Good zapping quality*, 3: *Fair zapping quality*, 2: *Poor zapping quality*, 1: *Bad zapping quality*.

The main result of [5] is an explicit relation between the user perceived QoE and the zapping time. From this relation it was deduced that in order to guarantee a MOS of at least 3.5, which is considered the lower bound for acceptable quality of service, see [6], we need to ascertain that Zapping

Time < 430 ms. Note that for MOS = 3.5 the average user will detect a slight degradation of the quality, of the considered service. The requirement on the zapping time mentioned above is currently not met in any implementation of IPTV, see for instance [7], and also subsection E in section 2. To increase the QoE of channel zapping, two approaches are possible. In the first approach the actual zapping time is reduced. An example of this method is given by Degrande et al. [8]. They suggest to retain the most recent video part in a circular buffer and display this video until the incoming channel is ready.

In the second approach the QoE is (possibly) increased by showing information while the user waits for the target channel to appear. The displayed information could be about the target channel, personalized content or advertisements, see also [9].

The aim of this paper is twofold. The first aim is to assess the QoE of channel zapping when, during zapping, advertisements are displayed, instead of the usual black screen. The second aim is to propose a system for optimal zapping experience. The system first estimates the zapping time (per target channel) and then, depending on this estimation, displays either a black screen, a picture or a clip.

The rest of this paper is organized as follows. In section 2, the possible effect of advertisements on IPTV perceived quality is analyzed and various factors that contribute to the results are listed. In Section 3 the experiment performed to quantify the user perception is described. In section 4, the results obtained from the subjective tests are presented. Section 5 describes a system for optimal zapping experience. Finally, conclusions are given in Section 6.

II. Quality of Experience and Advertisements

A. *Quality of Experience of IPTV*

Quality of Experience is the quality as judged by the user. QoE for IPTV is a subjective measure of the IPTV service that is evaluated by test subjects and depends on two types of factors. The first type of factors is due to the actual Quality of Service (QoS) or network quality being provided. The other type of factors result in a change of the user perception even though the QoS being provided remains the same.

Some of the factors that affect the QoE resulting from the actual QoS of the network are,

a) The zapping time;

b) The visual quality: this factor depends on the quality of encoding and decoding and on the packet loss in the network;

c) Synchronization between video and audio.

The other factors which result in variation in the user perception, even though the QoS remains the same, are:

a) The user device: the equipment the user is using to watch the channel is also important, for instance, the screen resolution of the TV;

b) The educational level, age and the TV watching experience of the customers;

c) The mood and concentration of the customer;

d) Viewing conditions, such as room illumination, display type (brightness, contrast), viewing distance etc.;

e) The IPTV service cost.

Measuring the QoE is very important for the service provider. Once the quality perceived by the user is measured, the vendor can determine the minimum requirements on the IPTV service quality (such as the maximum tolerable zapping time). Moreover, the vendors can provide additional services or use techniques to boost the user perception with the same QoS level being provided. For example, using advertisements during channel zapping may increase the QoE.

B. *Effect of Advertisements on QoE*

Using advertisements during the IPTV zapping times is an approach that tries to increase the QoE while the service quality or zapping time remains unchanged. Obviously, not all people would be happy to see advertisements during zapping. Therefore one could also think of educational or entertainment content during zapping. However, in this paper we focus only on advertisements because the business driver for this case is stronger. In fact, there are two major consequences that are expected to boost the QoE in the case of the actual implementation of this approach.

a) Users will watch the advertisements during the channel zapping, so they will not be bored with the longer zapping times. Hence, the perception of the user for the channels with advertisements could increase with respect to the black screens. This is actually what we have measured in the conducted subjective experiment.

b) The second consequence is that the providers will earn money from these advertisements. So, they can lower the price of the service. Obviously, a lower price is one of the factors that can boost the QoE.

It should be noted that the effect of advertisements on QoE is not just straightforward if it would be implemented. Rather, it depends on various factors which could affect the QoE positively or negatively.

a) The type of advertisement: A particular user could like some sort of advertisements and dislike other advertisements.

b) The content of the advertisement picture with respect to the length of the zapping time: For example, a glance of an advertisement that stays for a very short duration or a picture advertisement that stays static for a zapping time of 5 sec could be annoying for the user. Advertisements containing much text or video advertisements may be of little importance if zapping time is short.

c) The advertisement between the channels may need to be made random for better user perception; moreover, the advertisement set should be changed after some time.

Some of the factors above could positively affect the user perception. However, the implementation complexity also increases if all these issues are to be properly addressed. The best approach to use these advertisements is to select an advertisement randomly from a set of advertisements pre-rendered and stored in the Set-top Box (STB) when the user zaps to a different channel. Using pre-rendered advertisements is important because the zap screen can then be displayed immediately in this case.

III. The experiment

A. *Design of the experiment*

For the IPTV channel zapping experiment, a HTML page containing five animated gifs in different layers is implemented in JavaScript. These five animated gifs correspond to 5 different TV channel contents: an orchestra scene, two film trailers, a cartoon scene, and a sports scene. These animated gifs do not contain audio. Audio can be added but the synchronization problem will be another cause for quality degradation. So, to assess the quality experienced for zapping times, it is better to make the experiments with no sound, because otherwise the test subjects opinions might be biased by the synchronization quality. The animated gifs are displayed in a screen of size of 720x576 pixels in the HTML page. The page is designed in layers such that when the user zaps to a particular channel all animated gif layers become invisible except the layer containing the required animated gif.

In the experiment reported in [1], seven zapping times between 0 and 5 second were implemented in arrays in the javascript code. These zapping times were 0, 0.1, 0.2, 0.2, 0.5, 0.5, 1, 2, 2 and 5 sec. Some of the zapping times were repeated to see the consistency of the users. Moreover, a random ordering of these zapping times is implemented for each of the 12 test subjects that participated in the subjective experiment in [1]. When the user zaps to a new channel, the page sleeps for a time corresponding to the implemented zapping time before the requested channel is displayed. During this time a random advertisement picture is selected from a set of advertisement pictures and it is displayed. This chain of events is depicted in Fig. 1. For all advertisements we have used logo-like pictures.

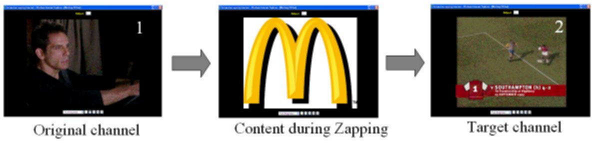


Figure 1. Showing an advertisement during zapping

When the user zaps to the next channel the same step is repeated, but an advertisement different from the advertisement shown during the previous zapping epoch is selected in random manner.

In order to increase the number of test subjects involved, we have set up an additional subjective experiment with 18 additional test subjects. This time the zapping times were 0.5, 1, 2, 3, 4 and 5 sec. The reason to include "new" zapping times (3 and 4 sec) is that we did not have acquired data for these zapping times. We left out some of the original zapping times in order to keep the length of the test sufficiently short, thus preventing fatigue of the test objects.

B. *The actual experiments*

In this subsection we list the outcomes of the original experiment with 12 test subjects and the additional experiments with 18 test subjects. The test subjects consisted of a total of 30 people at TNO ICT in Delft, the Netherlands. The test subjects varied in age, gender and experience.

To view the channels a laptop (Pentium 4, 2GB RAM, windows vista, 1500x750 pixels screen resolution) is used as a TV set. The experiment that we have conducted is of 'lean backward zapping' type. That means the user will sit back in a chair and use the remote control to zap between the channels. A Sony Ericsson Bluetooth enabled mobile phone is used as a remote control device. The experiment contains two parts, the training and the actual experiment.

In the training session, we show the test subjects three zapping times: instantaneous, intermediate and slow to give them an example of how the zapping times in the actual experiment are to be assessed. During this session, the test subject will get used to the ITU MOS scale.

During the actual experiments the test subjects were asked to experience the zapping times by zapping between the channels using the remote control (mobile phone) then to evaluate the experienced quality. For the test in [1] they evaluate their perception for the different zapping times, first using black screen and then using advertisements during zapping. For the new subjective test only the perception when using advertisements during zapping was asked. The users are also given the chance to give open suggestions. The handouts and scoring tables are given in Appendix A. To our knowledge these experiments were the first ever to assess the Quality of Experience when pictures are displayed during zapping.

IV. Results

A. *MOS results*

The results obtained for each zapping time are analyzed and averaged over the number of test subjects to obtain the MOS for each zapping time. This is done for both the case where a black screen is shown during zapping and the case where an advertisement is shown. From now on we will refer two these two cases as 'black screen' and 'advertisement'. The obtained results are shown in Table 1 and Table 2, respectively. Note that the results in Table 1 are completely based upon [1] while Table 2 is based upon [1] and the additional subjective experiment.

Table 1. MOS for 'black screen'

Zapping Time(sec)	MOS	Std. Dev.

0	4.75	0.62
0.1	4.83	0.39
0.2	4.42	0.67
0.2	4.25	0.62
0.5	3.33	0.89
0.5	3.50	0.67
1	2.75	0.75
2	1.83	0.83
2	1.83	0.58
5	1.08	0.29

Table 2. MOS for 'advertisement'

Zapping Time(sec)	MOS	Std. Dev.
0	4.42	1.24
0.1	2.92	1.38
0.2	3.04	1.20
0.5	3.40	1.27
1	3.37	0.85
2	2.71	0.97
3	2.17	1.04
4	1.61	0.85
5	1.77	1.01

As seen from the tables, the standard deviation is lower for the MOS of the black screen experiment. This implies the opinion of the users for the black screen zapping is quite stable. However, for an advertisement related MOS the opinion of different people shows more variance.

The MOS results, together with their 95% confidence intervals, are also shown in Fig. 2 and Fig. 3.

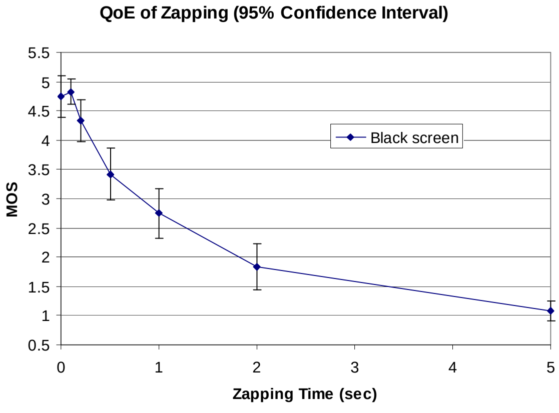


Figure 2. MOS for 'black screen'

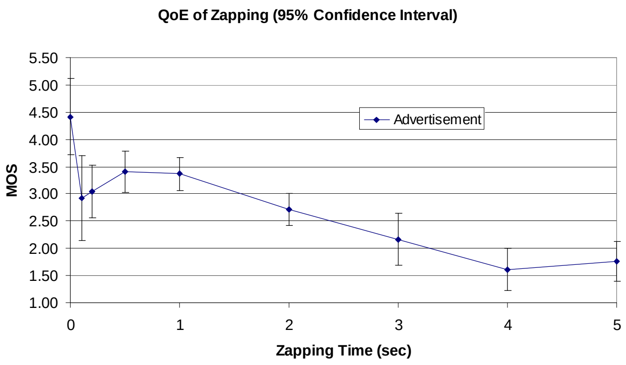


Figure 3. MOS for 'advertisement'

In order to compare the two cases, Fig. 4 contains the MOS results for both 'black screen' and 'advertisement'. The following important insights can be obtained from Figure 4:

- The QoE decreases as the zapping time increases, both for 'black screen' and 'advertisement', except for 'advertisement' for zapping times between 0.1 sec and 1 sec, and for 'advertisement' for zapping times between 4 sec and 5 sec.
- The MOS for 'advertisement' exceeds the MOS for 'black screen' for zapping times greater than 0.65 sec. This implies that the users prefer 'advertisement' only when the zapping time is sufficiently large. For zapping times of 1, 2, 3, 4 and 5 sec, the anticipated QoE increment is clearly seen, as the 'advertisement' curve for these zapping times shifts upwards with respect to the 'black screen' curve.
- The 'advertisement' MOS is more or less constant, for zapping times less than 1 sec. However, it decreases when the zapping time increases to 2 sec and 5 sec. This means users are still annoyed with longer zapping times, even though advertisements are shown during zapping.
- The QoE curve for 'advertisement' drops with high slope from zero zapping time to a zapping time of 0.1 sec. Because the 'black screen' curve decreases smoothly, we conclude that it is a bad idea to show advertisements in case of short zapping times.

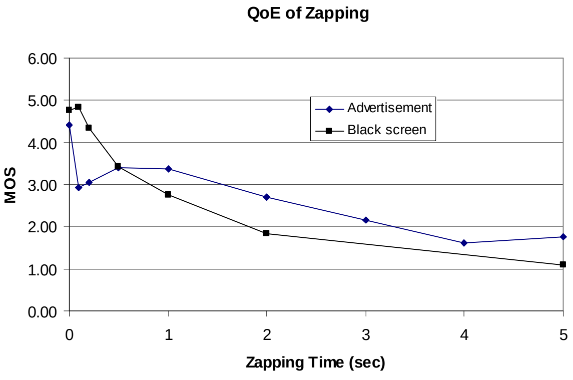


Figure 4. MOS for 'black screen' and 'advertisement'

B. *Comparison for 'black screen' with previous results*

The 'black screen' experiment was conducted before, see [5]. Our test scenarios are similar to the one reported in [5], except for some minor changes in the setup, like the laptop used for the experiment, the experiment room and the test subjects. The results obtained from the two tests are compared in the table below.

Table 3. 'Black screen' MOS for our experiment and previous experiment in [5]

Zapping Time(sec)	MOS Our experiment	MOS Experiment in [5]
0	4.75	4.90
0.1	4.83	4.90
0.2	4.42	4.60
0.2	4.25	4.50
0.5	3.33	3.50
0.5	3.50	3.30
1	2.75	2.30
2	1.83	1.60
2	1.83	2.00
5	1.08	1.10

It is clear that the outcome of the experiments is almost similar. In fact, the correlation between the two experiments is as high as 0.99.

The authors of [5] suggested the following model for the relation between zapping time (in sec) and QoE (expressed in MOS), for the 'black screen' case:

$$MOS = \max\{1, \min\{-1.02 \cdot \ln(ZappingTime) + 2.65, 5\}\} \tag{1}$$

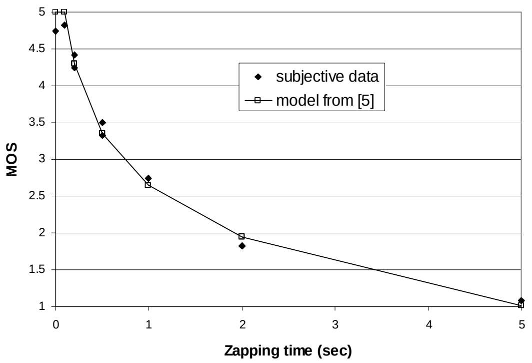


Figure 5. Comparing our 'black screen' results with the model from [5]

C. *QoE model for 'advertisement'*

Analogous to the QoE model for 'black screen' in [5] we will now suggest a QoE model for 'advertisement'. Using curve fitting on the following intervals (in seconds) for the zapping time: [0,0.1], and [0.1,5], we arrive at the following QoE model:

$$MOS = \max\{y_1, y_2\} \tag{2}$$

where

$$y_1 = -15 \cdot (ZappingTime) + 4.42 \,,$$

$y_2 = 0.0194x^5 - 0.2583x^4 + 1.307x^3 - 3.0864x^2 + 2.7366x + 2.6466$,

with x = Zapping Time (in seconds). Eq. (2) holds for Zapping Times on the interval [0, 5].

Using Table 2 we can validate the QoE model suggested in Eq. (2). This validation is visualized in Fig. 6.

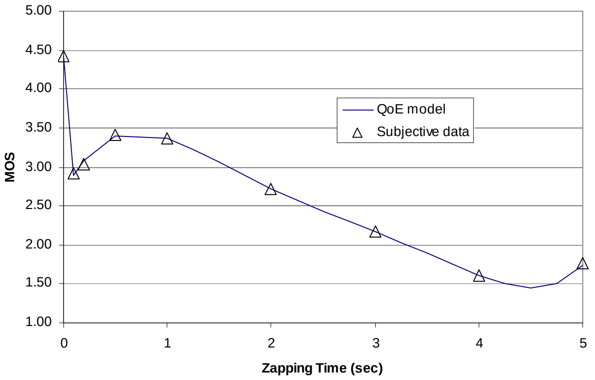


Figure 6 . MOS versus Zapping Time

It turns out that the correlation between the subjective data and the QoE model is 0.99 which is very high. In addition, the RMSE (Root Mean Square Error) equals 0.055 while the MCI (Mean Confidence Interval) satisfies 0.46. Therefore we conclude that the QoE model given by Eq. (2) is very useful for assessing the QoE of zapping for 'advertisement'.

D. *Discussion on user comments*

In addition to evaluating the MOS, users were asked to comment on the usability of advertisements during the zapping times and the reasons behind the MOS scores they gave.

The following are the main comments of the users,

a) A logo advertisement is not good enough for longer zapping times: Most users get annoyed with a single picture advertisement that is displayed for 2 or 5 seconds. It is better to put a video advertisement for such long zapping delays.

b) Advertisements which have darker (non-bright) colors are better for the user perception: A white background picture advertisement is not good if the channels have a black background. So, it is good to avoid dynamic changes in the frame color.

E. Typical zapping times

To obtain an indication of zapping times that are found in practice in today's digital television services, we performed some simple measurements. These measurements were carried out manually with the use of a stopwatch. For each of the services the test methodology used was the same. However it should be noted that testing of each of the services was performed only once, at one arbitrarily selected location during an arbitrary time of the day. Additionally, for each service, the measurements were carried out with the use of a single STB (Set-top Box). For most services it is possible to choose between several types of STB's. This means that none of the results necessarily is a good representation of the general performance of this service. However, in the assumption that providers aim to offer a constant quality level to all customers, these test results should give a reasonable indication for the overall service performance. The services that we have measured, all offered by Dutch providers, are:

- Provider 1: Digital TV service based upon DVB-T
- Provider 2: Digital TV service 1 over cable
- Provider 3: Digital TV service 2 over cable

In order to get a good indication of the overall zapping behavior of both services it is necessary to have a sufficiently large number of sample zapping times. For our test we switched channels in total 90 times. 45 times we used the arrow buttons to zap sequentially from channel 1 to 10 and back to channel 1. Subsequently we followed the same procedure to measure zapping delays while using the number buttons of the remote control. Since we did the tests manually with a stopwatch, the tests were performed with the volume of the TV turned off (muted). In this way the person performing the measurement would not be influenced by the sound that belongs to the image (sometimes "sound switching times" show different zapping behavior than the "image switching times").

We limited the test between channels 1 to 10 because all three services in this test offered the "main channels" on these channel numbers. These channels were; NED1, NED2, NED3, RTL4, RTL5, SBS6, RTL7, Veronica/Jetix, NET5 and RTL8.

To measure the zapping time between channels we started the stopwatch simultaneously with the push of the button on the remote control (in case we zapped to e.g. channel 14 we started the stopwatch at the moment that the button for the second figure was pushed, in this case 4). As soon as we saw an image we stopped the time. Then we subtracted 0.2 seconds from this time in order to compensate for the human response delay.

In Table 4, the mean delays and the variations in zapping delay are listed for all measured services.

Table 4: Measured zapping times from existing services

Service	Provider 1	Provider 2	Provider 3
STB type	Samsung SMT-1000T	Thomson 52UPC01	Humax IR-Fox C
Arrows			
Mean delay	2,12 s	1,30 s	1,82 s
Variation	0,44 s	0,09 s	0,23 s
Numbers			
Mean delay	3,65 s	1,36 s	4,45 s
Variation	0,33 s	0,11 s	0,25 s
Combined			
Mean delay	2,89 s	1,33 s	3,14 s
Variation	0,97 s	0,10 s	1,99 s

Notice the large differences between the average zapping times for "zapping by arrows" and "zapping by numbers" in the Provider 1 and Provider 3 case. These differences are caused by the implementation of the STB's. When using the arrows, the STB starts fetching the new channel immediately. When pushing a number, the STB "waits" a short period for an additional number. For example: if a user pushes number 1 on the remote control in order to zap to channel 1, then the STB waits a short period for an additional number in case a user intends to zap to a different channel. This could be e.g. number 2 to zap to channel 12 or number 3 to zap to channel 13. Only after this short period, if the second number is not detected, then the STB starts fetching channel 1. From the differences between the averages we can conclude that this "waiting period" is about 1.5 seconds in the Provider 1 STB implementation and about 2.5 seconds in the Provider 3 STB implementation.

V. System for optimal zapping experience

From our conducted subjective experiment, it is found that, for some ranges of zapping times, advertisements lead to a better QoE than black screens. However, for small zapping times black screen is found to be better, see also Fig. 4. However, users also commented that for longer zapping times, i.e. in the order of at least 2 seconds, a video advertisement is preferred over a picture advertisement. Thus it can be anticipated that in order to compare the QoE of 'black screen', 'still picture' and 'clip', Fig. 7 can be used.

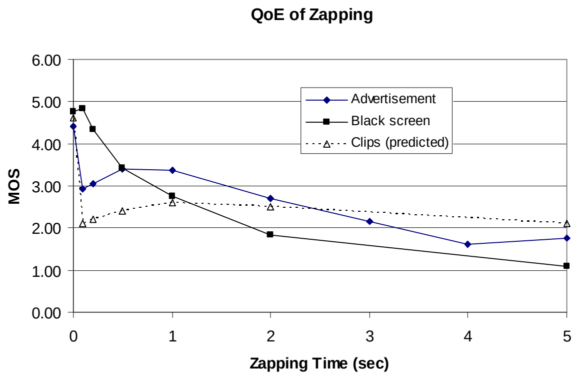


Figure 7. MOS for 'black screen', 'still picture' and 'clip'

Note that the MOS curve for 'clip' in Fig.7 is an anticipated prediction; it is not based upon actual subjective experiments with 'clips' during channel zapping. Based upon Fig.7, we propose a system for optimal zapping experience.

A. Requirements

In the IPTV provider network usually not all streams originate from the same source. The most popular ones are streamed from the 'edge' of the network so as to enhance startup times. Less popular channels are streamed from a more central location as this reduces bandwidth consumption and hence reduces cost [10]. The system is subject to a number of imminent changes in the environment, e.g. channels that change the source of origin, new home network equipment and a change of service provider, etcetera. A system like this one needs to adapt to such changes itself, in an automated manner.

Related, but slightly different is the requirement that the system must be zero-configuration. A STB is a device with limited user interfacing capabilities. Configuring one with only a remote control is a daunting task for most regular users already, so having users configure initial settings regarding the ISP or SP network settings should be avoided whenever possible.

Many other requirements exist, but these are considered out of scope.

B. System Overview

The main task of the system is to first estimate the zapping time and then, depending on this estimation, display either a black screen, a picture or a clip. As we learn from the requirements above, its secondary task is to constantly improve the prediction.

Three main functional blocks can be distinguished within the system: *(i)* the Set-top Box (STB) functionality, *(ii)* the Delay Predictor and *(iii)* the Content Selector. These components work together to fulfill the two tasks stated above. Fig. 8 depicts the system overview. The system boundary is indicated with a dashed line.

All messages referenced in the text below refer to the numbered arrows in this figure. Arrows with filled solid heads indicate that a result is received in response to the corresponding message. E.g. message 2.2 actually returns a value; the delay. It is therefore required that the Delay Predictor itself finishes the calculation of that value. During this processing time the message sender must wait. This behavior is referred to as blocking, or synchronous. Arrows with stick heads on the other hand indicate asynchronous messages. The sender does not need to wait for the receiving module to process the message and come up with a result.

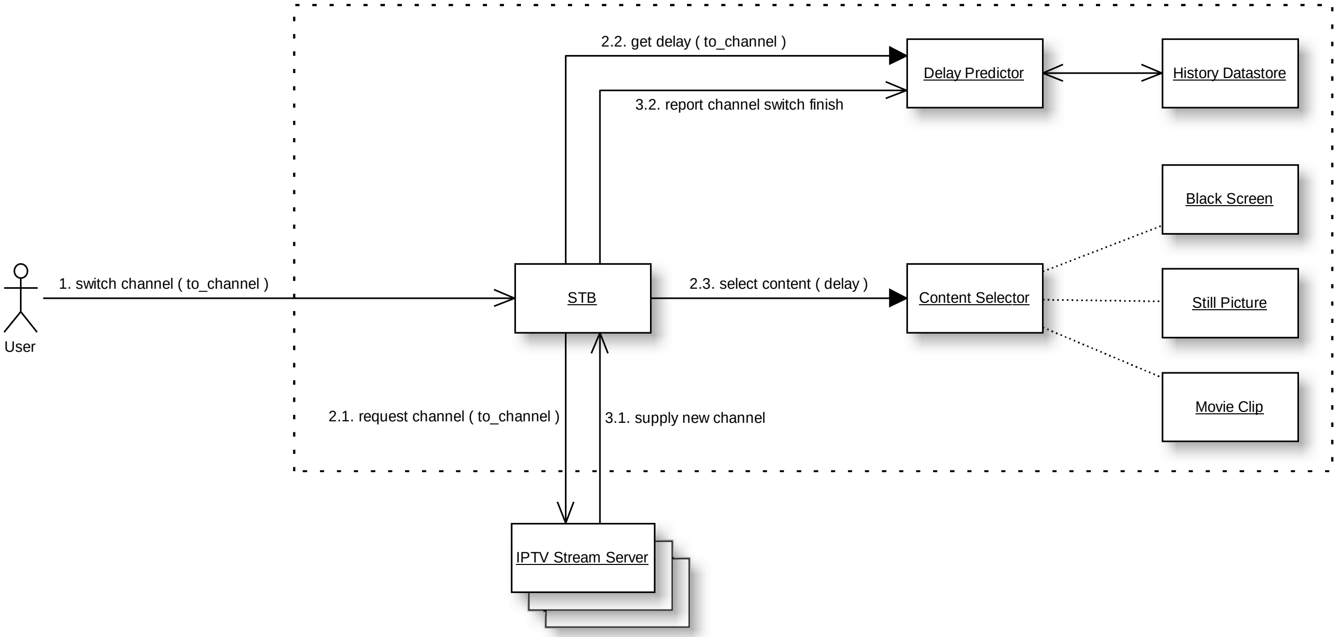

2.2. get delay (to_channel)
Delay Predictor
History Datastore
3.2. report channel switch finish
Black Screen
User
1. switch channel (to_channel)
STB
2.3. select content (delay)
Content Selector
Still Picture
Movie Clip
2.1. request channel (to_channel)
3.1. supply new channel
IPTV Stream Server

Figure 8. Overview of System for Optimal Zapping Experience

Instead, the sending party treats such messages as “fire and forget”.

C. *Selection*

The STB comprises all functionality of a regular STB. It listens for user requests, e.g. by means of an infrared receiver and a remote control. It requests video streams from external sources, e.g. through multicast joins and it renders the picture to an external television set.

In the presented system it also manages the in-between content. It does so by first querying the Delay Predictor to get the delay for the user-selected channel (message 2.2). The returned value is the estimated waiting time or delay, which is forwarded to the Content Selector (message 2.3). The Content Selector determines the recommended action to be taken to optimize the waiting experience. The set of actions is a mapping from a time interval to a preferred action, as shown in the table below.

Table 5. Mapping of Time interval to Action

Time interval (seconds)	**Action**
$0 < \tau \leq a$	Do nothing
$a < \tau \leq b$	Show black screen
$b < \tau \leq c$	Show picture
$c < \tau \leq d$	Show video clip
$\tau > d$	Show sequence of video clips

In this table values *a* through *d* depict time values, where $0<a<b<c<d$ and τ represents the predicted time.

Finally, the determined action is returned in response to message 2.3., accompanied by a reference to the content to use. The STB displays the indicated content item until the new channel stream is ready to be displayed (message 3.1).

D. *Adjusting the prediction*

In order to be able to adjust its future estimations, the Delay Predictor must be aware of the actual duration of channel switches. A new waiting time measurement is triggered by the “2.2 get delay” message. The time instance this message is received is recorded as T_0. The time measurement ends when a “report channel switch finish” message is received. The time instance this event is received is recorded as T_1. Based on T_0 and T_1 the waiting time for this request is determined:

$$T_{waiting\_time} = T_1 - T_0 \, . \tag{3}$$

The measurement is stored in the History Datastore, along with an identifier for the request that comprises the target channel (`to_channel`) which can be a Uniform Resource Identifier (URI).

E. *Prediction*

Using the identifier for the user request as presented above, a prediction is made for the expected waiting time from previous measurements for the same request (requests with the same identifier). It is assumed here that the expected delay is dependent only on the newly selected channel. The prediction is a default value when no measurements are available, otherwise it is the exponential weighted average like:

$$S_t = (1-\alpha)\cdot S_{t-1} + \alpha \cdot x_t \tag{4}$$

where S_t is the estimation, S_{t-1} is the previous estimation, α is the smoothing constant and x_t the last measurement.

While being simple to implement on a limited resources device like a STB, this still provides a zero-configuration solution that adapts well to a changing environment.

F. *Optional Changes*

- The prediction as performed by the Delay Predictor may be based on other parameters like the time of day, the number of concurrent users (the system load) or anything else that could influence the waiting time.
- The system may be adapted as to provide an ‘optimal waiting experience’, i.e. not only for use in IPTV environments but also in ATMs, web browsers and any other situation where a user has expressed a wish to some machine while the waiting time is non-deterministic.

VI. CONCLUSIONS

Measuring the QoE of IPTV is an important issue for vendors and service providers. Channel zapping time is a major factor that affects QoE in IPTV. One of the ways to increase the user perceived quality of channel zapping, is to display advertisements during zapping, instead of the usual black screens. From our conducted subjective experiment, it is found that, for some ranges of zapping times, advertisements lead to a better QoE than black screens. However, for small zapping times black screen is found to be better. For intermediate zapping times picture advertisements are convenient. For longer zapping times picture advertisements give a better QoE than black screens; however, using video advertisements might give even better QoE in that situation. In the future we would like to conduct subjective tests where advertisement clips are used during long zapping times. This work might lead to the establishment of two zapping time thresholds: a black screen should be used below the lower threshold and video advertisements above the higher threshold. We also plan to conduct subjective tests in other countries, to see whether or not regional differences occur.

Lastly, we have shown how to implement a ‘System for Optimal Zapping Experience’. In the near future we will implement the system in a field trial and will conduct further subjective experiments with the system, taking into account longer time scales, i.e. we will assess QoE after a period of for instance 3 months. Finally we suggested that the system might be adapted to function as a broader ‘System for Optimal Waiting Experience’. Hopefully this inspires others to improve the quality of experience in our day to day activities.

ACKNOWLEDGMENT

This work is partially supported by the CELTIC Project RUBENS. For more information, see http://wiki-rubens.celtic-initiative.org.

REFERENCES

[1] B.E. Godana, R.E. Kooij and O.K. Ahmed, Impact of Advertisements during Channel Zapping on Quality of Experience, Proc. of The Fifth International Conference on Networking and Services, ICNS 2009, Valencia, Spain April 20-25, 2009.

[2] Spirent communications white paper, “Delivering optimal QoE for IPTV success”, Feb.2006.

[3] ITU Focus group on IPTV, “Quality of Experience requirements for IPTV”, FG IPTV-DOC-0118, July 2007.

[4] DSL Forum, “Triple Play Services Quality of Experience (QoE) Requirements and Mechanisms”, Technical Report TR-126, December 2006.

[5] Robert Kooij, Kamal Ahmed, Kjell Brunnström, “Perceived Quality of Channel Zapping”, Proc. of the fifth IASTED International Conference, Comm. Systems and Networks, Aug. 28-30, 2006, Palma de Mallorca, Spain, pp. 155-158.

[6] ITU-T Rec. P.800, “Methods for Subjective, Determination of Transmission Quality”, July 1996.

[7] Nick Fielibert, “Quality Issues in IP Delivery: Set-top boxes”, September 2008, http://www.ieee.org/organizations/society/bt/08iptv3.pdf

[8] N. Degrande, K. Laevens, D. De Vleeschauwer, R. Sharpe, “Increasing the user perceived quality for IPTV services”, IEEE Communications Magazine, Vol. 46, Issue 2, Feb. 2008, pp. 94-100.

[9] Bigband Networks, “Methods and apparatus for using delay time during switching events to display previously stored information elements”, US Patents 7237251, March 2000.

[10] J. Caja, “Optimization of IPTV Multicast Traffic Transport over Next Generation Metro Networks,” in Telecommunications Network Strategy and Planning Symposium, 2006. NETWORKS 2006. 12th International, Nov. 2006, pp. 1–6.

APPENDIX A: QUESTIONAIRE FOR SUBJECTIVE ASSESMENT

Objective: To assess the user perceived quality of digital TV when displaying advertisements during zapping.

General introduction

Nowadays various digital television services are available. But, the user of digital TV experiences longer time delays than the traditional analog TV while switching from channel to channel. The digital TV operator usually displays a black screen during these switching times. However, it could be interesting for the user of digital TV, if the operator displays advertisements instead of a black screen. In this experiment we want to quantify the quality the end user perceives when advertisements are displayed during these switching times.

Introduction to the experiment

In the following experiments you will be asked to assess a total of 10 switching times. The first set of experiments will be done for a black screen and the second set of experiments with advertisements. For this purpose five pre-programmed TV channels are used. You can switch between the channels by pressing the keys 1 to 5 on the mobile phone (“the remote control device”). To change to a different switching time, use the volume up/down keys on the mobile phone. The task is to assess the duration of these switching times using Mean Opinion Score values shown in the table.

Mean Opinion Score:	Explanation:
5	Excellent
4	Good
3	Fair
2	Poor
1	Bad

Training

Three switching times are shown in the training. These switching times will be rated as shown in the following table.

Switching Time	MOS
instantaneous	5
intermediate	between 5 and 1
slow	1

The actual experiment

Follow the instructions below for both Set 1 (black screen) and Set 2 (advertisement) experiments.

1. In the opening screen: select your subject number.
2. Select “switching time 1” in the drop-down list on the bottom of the screen.
3. Experience the switching time and write down the MOS value in the table for this switching time.
4. Then select “switching time 2” in the drop-down list (can also be changed using the volume up/down button of the mobile phone) and repeat step 3 until you have assessed all 10 switching times.

Usage note:

Action	Key to use on mobile
Zap between channels	1 to 5
Change switching time	volume up/down

Form:

	Black screen MOS	**Advertisement MOS**
1		
2		
3		
4		
5		
6		
7		
8		
9		
10		

Further Suggestions:

1. …

2. …

Localization in WiMAX Networks Depending on The Available RSS-based Measurements

Mussa Bshara, *Student Member, IEEE,* and Leo Van Biesen, *Senior Member, IEEE*

***Abstract*—Recently, localization in wireless networks has gained a lot of interest; especially after some of the most interesting positioning application areas have emerged in wireless communications. The most important are the Federal Communications Commission (FCC) and the European Recommendation E112, both of which require that wireless providers should be able to locate within tens of meters users of emergency calls.**

In this paper, fingerprinting-based localization depending on the available RSS-based measurements has been addressed in WiMAX networks. Fingerprinting is a good way to overcome signal propagation peculiarities caused by the propagation environment. And using the available RSS-based measurements is of great interest because of their availability during the normal operation of the standard modems, and the simplicity of obtaining them. The obtained results show that using the available RSS-based measurements to locate users in WiMAX networks is feasible and provides the required positioning accuracy for most of location-based services (LBS), if fingerprinting-based approaches are used to obtain localization. The available RSS-based measurement in the current WiMAX modems is called *SCORE*. The term SCORE is used by modem manufacturer to indicate the quality of the connection between the base station (BS) and the subscriber station (SS). However, using the actual received signal strength (RSS) values gives higher accuracy than using SCORE values, but obtaining them is more difficult and not feasible using the current modems.

***Keywords*-Fingerprinting, GPS, GSM, location-based services, localization, positioning, positioning accuracy, power maps , received signal strength, SCORE, WiMAX.**

I. INTRODUCTION

THERE are several ways to position a wireless network user. Global positioning system (GPS) is the most popular way, it provides positioning accuracy that meets all the known location-dependent applications requirements. The main problems with GPS -despite that the user's terminal must be GPS enabled- are the battery high consumption, the limited coverage and the latency. The battery high consumption means that the user can be positioned during a short period of time. Also GPS performs poorly in urban areas near the high risings and inside the tunnels, i.e. it has a poor performance when it is needed the most. And it needs about 4 minutes (cold start) before the first position fix is available. Another way to position a user, is to depend on the wireless network itself by using the available information such as SCORE information [1], or Cell-ID which has been used widely in global system for mobile communications (GSM) despite its limited accuracy [2]. One can also make measurements on the network to obtain localization, some of these measurements are hard to obtain such as time of arrival (TOA) which needs synchronization, and some are easy to obtain such as RSS measurements [3], [4]. Adding new hardware to the base stations (BSs) to improve the measurements accuracy (for example using array antennas or adding localization measurement unit (LMU) to each base station) can provide high localization accuracy, but this option suffers from the high roll-out cost. Some of the measurements can be conducted by the terminal itself, others can be only obtained by the network. From now on, we will refer to the measurements as network measurements regardless where these measurements have been conducted; in the network itself (network side), in the user terminal (terminal side) or in both. Many localization approaches depending on network measurements have been proposed in GSM networks and sensor networks. Most of the work focused on range measurements depending on TOA, time-difference of arrival (TDOA) and RSS observations, surveys [3], [5] and [6]. These approaches improved potentially the localization accuracy achieved by using the Cell-ID. The exponential path loss model which is known as the Okumura-Hata (OH) model [7], [8] was the first approach to be used to obtain positioning. In [9] the authors propose using statistical log-normal model of RSS measurements and the sequential Monte Carlo localization technique, to get better localization accuracy. In [10] an enhanced object tracking with RSS using Kalman Filter is proposed by obtaining velocity information of the mobile sensor node which is used to improve the accuracy of the tracking. The authors of [11] proposed using a grid-based centralized localization using RSS to locate a target using the maximum and minimum path loss exponents. The proposed method achieves higher localization accuracy than the conventional localization method using the same path loss exponent when the distribution of the path loss exponents over the field is uniform, and has worse performance when the distribution of the path loss exponents over the field is normal distribution. An indoor localization method based on received signal strength using discrete fourier transform has been proposed in [12]; the method provided satisfactory positioning accuracy if the environment stay consistent from the radio building phase. The fingerprinting approach which depends on comparing the on-line measurements obtained by the user with an already built database, proved to provide better performance than OH alternative [2]. In [13],[14] the authors used fingerprint approach to mitigate the effect of multipath components and the inconveniences related to OH approach to provide better positioning accuracy.

In this paper, we propose using fingerprinting-based local-

Mussa Bshara and Leo Van Biesen are with the Department of Fundamental Electricity and Instrumentation, Vrije Universiteit Brussel, Brussels, Belgium (e-mail: {mbshara,lvbiesen}@vub.ac.be).

ization depending on SCORE observations for positioning in WiMAX networks. The importance of the contributions of this paper can be summarized as follows:

- This paper provides a detailed description of all the RSS-based quantities that can be measured using the current WiMAX networks.
- The used RSS-based quantity, which is the SCORE, is new and has never been used before in any localization approach in wireless networks.
- Introducing an important enchantment to the classical fingerprinting approach based on the fact that the Cell-IDs are less affected by the propagation environment than RSS-based values.

We also argue that this approach meets the requirements of most of the known LBS such as discovering the nearby places, whereabouts of a friend, user tracking and many other services.

This paper is organized as follows: Section II discusses the positioning possibilities in WiMAX networks. Section III discusses the RSS-based measurements in the current WiMAX networks. The fingerprinting-based localization approach depending on RSS-based measurements is discussed in section IV, and we conclude in section V.

II. Positioning possibilities in WiMAX networks

The topology of WiMAX network is similar to GSM one; the two of them use base stations to establish a wireless connection with subscriber stations (GSM terminal or WiMAX enabled computer for example). And almost the same quantities can be measured using both networks. In WiMAX networks the following quantities can be used for possible localization application:

- *The Timing Adjust (TA):* This concept is similar to timing advance (TA) or time of arrival (TOA) concept in GSM networks [15].
- *The Time Difference of Timing Adjust (TDOTA):* This concept is similar to the time difference of arrival (TDOA) concept in GSM networks. The idea of this measurement is to compare more than one TA values measured to different base stations to eliminate the measurement error caused by the terminal clock synchronization as long as the network is synchronized.
- *The Angle of Arrival (AOA):* WiMAX uses directional antennas which allow the determination of the azimuth of a terminal seen by a certain base station. The current antennas used in Pre-WiMAX network in Brussels provide this information as sectors (60 , 90 and 120 degrees). WiMAX networks started to use advanced antenna arrays where *beamforming* allows rotating narrow beams. The narrow antenna patterns will increase the accuracy of the measured terminal azimuth.
- *The Base Station Identifier (BSID):* This concept is the same as Cell-ID in GSM networks. The position of a terminal can be determined depending on the serving base station coordinates. This value can be obtained -by the terminal- by obtaining the serving base station MAC address which is broadcasted over the control channel.
- *The Received Signal Strength Index (RSSI):* WiMAX terminals can measure the received power broadcasted by a base station (Extra software needed). This measurement gives information about the distance between the terminal and the corresponding base station. The RSSI values depend on the operating environment and a path loss model has to be developed for a certain environment.
- *The SCORE values:* The current standard WiMAX terminals measure the SCORE values of the available BSs. The SCORE values are related directly to the RSSI values. However, they can be considered -with some approximations- as rough RSSI measurements.

In addition, the support of short-range communications among the terminals (mesh networks) was proposed in WiMAX networks [15]. The rationale for introducing short-range communications is mainly due to three arguments:

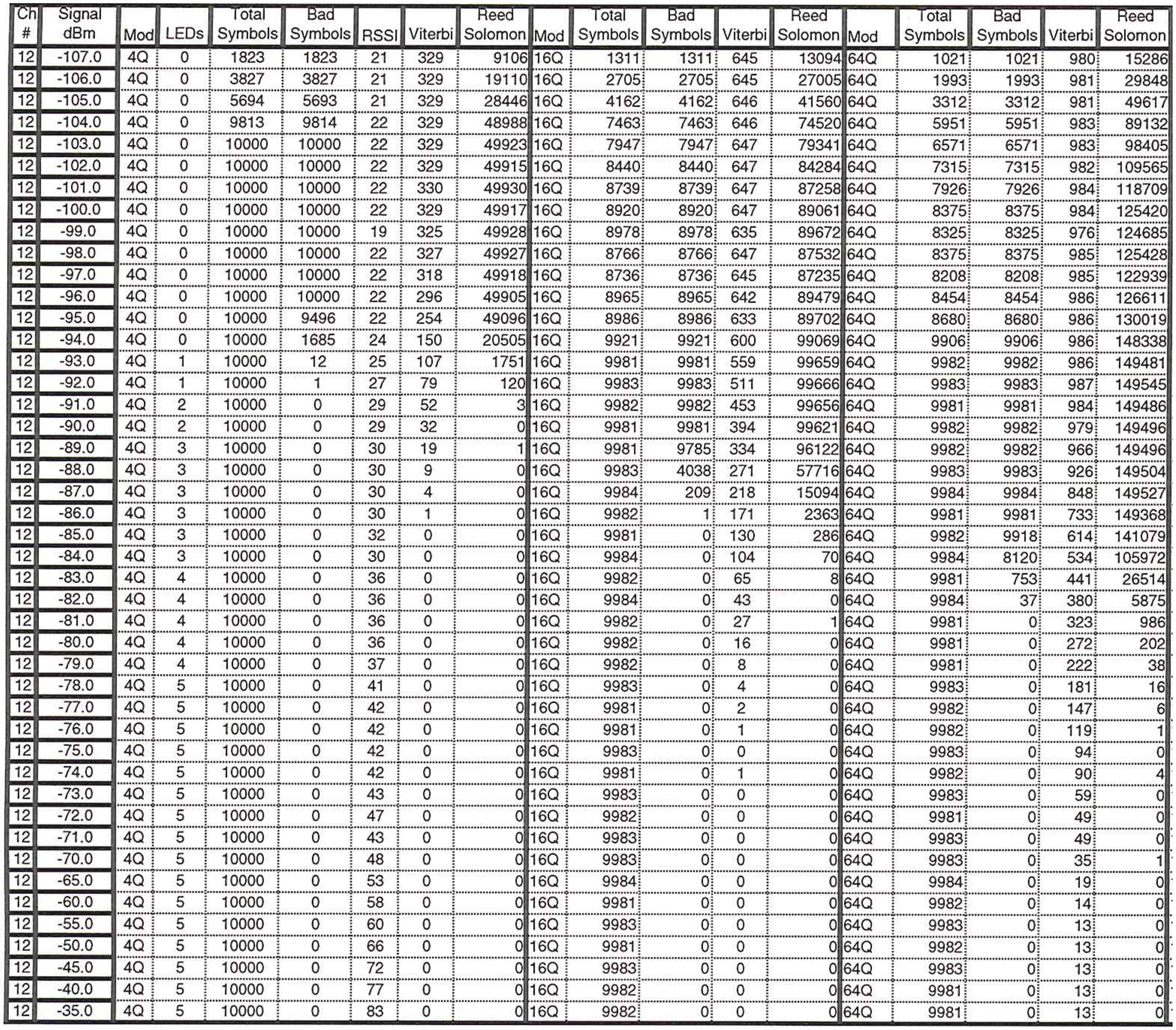
1) The need to extend the coverage to places not covered by a base station.
2) Support peer-to-peer (P2P) high-speed wireless links between the terminals.
3) The need to enhance the communication between a terminal and the base station by fostering cooperative communication protocols among spatially proximate devices.

The accuracy of the location estimation can be enhanced by utilizing the additional information gained from measuring the relative distances between the terminals. The support of short-range communications is very attractive, but the practical implementation is very complicated and has a lot of complications. Therefore, the use of mesh networks could be avoided or be limited to security and emergency cases. For example, a police car (or an ambulance) can establish a direct connection to other cars; or in case of being outside the coverage area of the wireless network, a connection can be established to the main network backbone by using the available modems in its range.

Therefore, WiMAX networks have all the resources to locate their subscribers without relying on any external system. The most attractive resources for localization are the ones that are easy to obtain and already available during the normal terminal operation such as RSS-based values.

III. The RSS-based measurements in the current WiMAX networks

In this section, we consider the received power measurements. The received power is usually measured in watts (or dBm) and can be obtained by using special equipments such as power meters, base station analyzers or spectrum analyzers. In many applications the use of the mentioned equipments (or similar ones) is not possible due to many different reasons including the size and weight, the power consumption, the usage complications and the price. In real life, the choice of a certain measurement equipment depends on the application requirements, i.e., the received power could be measured using



Ch #	Signal dBm	Mod	LEDs	Total Symbols	Bad Symbols	RSSI	Viterbi	Reed Solomon	Mod	Total Symbols	Bad Symbols	Viterbi	Reed Solomon	Mod	Total Symbols	Bad Symbols	Viterbi	Reed Solomon
12	-107.0	4Q	0	1823	1823	21	329	9106	16Q	1311	1311	645	13094	64Q	1021	1021	980	15286
12	-106.0	4Q	0	3827	3827	21	329	19110	16Q	2705	2705	645	27005	64Q	1993	1993	981	29848
12	-105.0	4Q	0	5694	5693	21	329	28446	16Q	4162	4162	646	41560	64Q	3312	3312	981	49617
12	-104.0	4Q	0	9813	9814	22	329	48988	16Q	7463	7463	646	74520	64Q	5951	5951	983	89132
12	-103.0	4Q	0	10000	10000	22	329	49923	16Q	7947	7947	647	79341	64Q	6571	6571	983	98405
12	-102.0	4Q	0	10000	10000	22	329	49915	16Q	8440	8440	647	84284	64Q	7315	7315	982	109565
12	-101.0	4Q	0	10000	10000	22	330	49930	16Q	8739	8739	647	87258	64Q	7926	7926	984	118709
12	-100.0	4Q	0	10000	10000	22	329	49917	16Q	8920	8920	647	89061	64Q	8375	8375	984	125420
12	-99.0	4Q	0	10000	10000	19	325	49928	16Q	8978	8978	635	89672	64Q	8325	8325	976	124685
12	-98.0	4Q	0	10000	10000	22	327	49927	16Q	8766	8766	647	87532	64Q	8375	8375	985	125428
12	-97.0	4Q	0	10000	10000	22	318	49918	16Q	8736	8736	645	87235	64Q	8208	8208	985	122939
12	-96.0	4Q	0	10000	10000	22	296	49905	16Q	8965	8965	642	89479	64Q	8454	8454	986	126611
12	-95.0	4Q	0	10000	9496	22	254	49096	16Q	8986	8986	633	89702	64Q	8680	8680	986	130019
12	-94.0	4Q	0	10000	1685	24	150	20505	16Q	9921	9921	600	99069	64Q	9906	9906	986	148338
12	-93.0	4Q	1	10000	12	25	107	1751	16Q	9981	9981	559	99659	64Q	9982	9982	986	149481
12	-92.0	4Q	1	10000	1	27	79	120	16Q	9983	9983	511	99666	64Q	9983	9983	987	149545
12	-91.0	4Q	2	10000	0	29	52	3	16Q	9982	9982	453	99656	64Q	9981	9981	984	149486
12	-90.0	4Q	2	10000	0	29	32	0	16Q	9981	9981	394	99621	64Q	9982	9982	979	149496
12	-89.0	4Q	3	10000	0	30	19	1	16Q	9981	9785	334	96122	64Q	9982	9982	966	149496
12	-88.0	4Q	3	10000	0	30	9	0	16Q	9983	4038	271	57716	64Q	9983	9983	926	149504
12	-87.0	4Q	3	10000	0	30	4	0	16Q	9984	209	218	15094	64Q	9984	9984	848	149527
12	-86.0	4Q	3	10000	0	30	1	0	16Q	9982	1	171	2363	64Q	9981	9981	733	149368
12	-85.0	4Q	3	10000	0	32	0	0	16Q	9981	0	130	286	64Q	9982	9918	614	141079
12	-84.0	4Q	3	10000	0	30	0	0	16Q	9984	0	104	70	64Q	9984	8120	534	105972
12	-83.0	4Q	4	10000	0	36	0	0	16Q	9982	0	65	8	64Q	9981	753	441	26514
12	-82.0	4Q	4	10000	0	36	0	0	16Q	9984	0	43	0	64Q	9984	37	380	5875
12	-81.0	4Q	4	10000	0	36	0	0	16Q	9982	0	27	1	64Q	9981	0	323	986
12	-80.0	4Q	4	10000	0	36	0	0	16Q	9982	0	16	0	64Q	9981	0	272	202
12	-79.0	4Q	4	10000	0	37	0	0	16Q	9982	0	8	0	64Q	9981	0	222	38
12	-78.0	4Q	5	10000	0	41	0	0	16Q	9983	0	4	0	64Q	9983	0	181	16
12	-77.0	4Q	5	10000	0	42	0	0	16Q	9981	0	2	0	64Q	9982	0	147	6
12	-76.0	4Q	5	10000	0	42	0	0	16Q	9981	0	1	0	64Q	9982	0	119	1
12	-75.0	4Q	5	10000	0	42	0	0	16Q	9983	0	0	0	64Q	9983	0	94	0
12	-74.0	4Q	5	10000	0	42	0	0	16Q	9981	0	1	0	64Q	9982	0	90	4
12	-73.0	4Q	5	10000	0	43	0	0	16Q	9983	0	0	0	64Q	9983	0	59	0
12	-72.0	4Q	5	10000	0	47	0	0	16Q	9982	0	0	0	64Q	9981	0	49	0
12	-71.0	4Q	5	10000	0	43	0	0	16Q	9983	0	0	0	64Q	9983	0	49	0
12	-70.0	4Q	5	10000	0	48	0	0	16Q	9983	0	0	0	64Q	9983	0	35	1
12	-65.0	4Q	5	10000	0	53	0	0	16Q	9984	0	0	0	64Q	9984	0	19	0
12	-60.0	4Q	5	10000	0	58	0	0	16Q	9981	0	0	0	64Q	9982	0	14	0
12	-55.0	4Q	5	10000	0	60	0	0	16Q	9983	0	0	0	64Q	9983	0	13	0
12	-50.0	4Q	5	10000	0	66	0	0	16Q	9981	0	0	0	64Q	9982	0	13	0
12	-45.0	4Q	5	10000	0	72	0	0	16Q	9983	0	0	0	64Q	9983	0	13	0
12	-40.0	4Q	5	10000	0	77	0	0	16Q	9982	0	0	0	64Q	9981	0	13	0
12	-35.0	4Q	5	10000	0	83	0	0	16Q	9982	0	0	0	64Q	9981	0	13	0

Fig. 1: The modem calibration table (used to convert the RSSI values to RSS ones measured in dBm).

different equipments for different purposes or applications. For example it could be measured using a small sized and simple equipment, if the measurement accuracy is enough for a certain decision or application. Take for example a GSM terminal, it continuously measures the received power from the neighboring base stations. It is true that the measurement accuracy is not that high, but it is certainly enough to decide the best base station. The same thing applies to WiMAX modems, they also measure the received power and they use this measurement to decide the quality of the connection between the modem and the neighboring base stations. In this research, we distinguish between three types of RSS-based measurements:

1) *The Received Signal Strength* (RSS): The RSS values represent the actual measured power (in dBm). Usually they are presented as real values with 2 decimal digits resolution, however they can be found as integer values depending on the used measurement equipment. These values are measured using dedicated equipments such as spectrum analyzers.
2) *The Received Signal Strength Index* (RSSI): The RSSI values are presented as positive integer values. They are

obtained by WiMAX modems and are presented only for the serving BS.

3) *SCORE :* The SCORE values are obtained by the standard WiMAX modems simultaneously for all the available base stations and presented as positive integer values.

In this research, only the quantities that can be measured by WiMAX modems are used; i.e, RSSI and SCORE values. And all the used values were converted to RSS values measured in dBm as explained in sections III-A and III-B.

A. Converting RSSI values to RSS values

To obtain the actual received power measured in dBm from RSSI measurements, a calibration table has to be used. For each used modem (in measurements), a calibration table is generated. Figure 1 shows the used calibration table in our measurements (only channel 12 can be found in the provided table, but in the original table channels 2 and 27 can be also found). The table contains information about the measured WiMAX signals, such as channel number (frequency), coding, *Viterbi* decoder information etc ..., and it also contains the equivalent RSS values to a set of RSSI ones. The conversion from RSSI to RSS is done by searching the calibration table for the equivalent RSS value to a certain RSSI one.

B. Converting SCORE values to RSS values

There are no available calibration tables that give the equivalent RSS values to SCORE measurements. Therefore, the solution is to convert the SCORE values to RSSI ones using the equation 1, and then using the calibration tables to obtain the equivalent RSS values in dbm. The relation between SCORE and RSSI values according to the information provided by the modem manufacturer is given by the following equation:

$$SCORE = (RSSI - 22) - (0.08 \times AvgViterbi) \qquad (1)$$

Where, *AvgViterbi* is a value generated by the modem decoder.

C. Practical issues on obtaining RSS from RSSI and SCORE values

Using calibration tables directly to convert RSSI values to RSS is not practical due to the following reasons:

1) Converting a large number of values needs relatively long time to look up the tables to find the equivalent RSS values to the measured RSSI ones. The processing power is an issue in portable devices solutions.
2) Finding the exact RSSI values in the calibration table is not guaranteed. The calibration table is generated for a set of RSSI values, so it is not guaranteed to find all the measured values in the table. In many cases, only close values to the exact ones can be found.
3) It is possible - in few cases- to find more than one equivalent RSS value to the same RSSI measurement. For example, in the table shown in figure 1, the equivalent RSS values to the RSSI value of 21, are: -107 dBm, -106 dBm and -105 dBm. And for the RSSI value of 22, the equivalent RSS values are: -104 dBm, -103 dBm, -102 dBm, -101 dBm and -100 dBm.

TABLE I: The fitting curves coefficients for channels 2, 12 and 27.

	Channel 2	Channel 12	Channel 27
p1	-6.434e-006	-8.381e-006	-8.744e-006
p2	0.001493	0.001817	0.001879
p3	-0.1252	-0.1426	-0.144
p4	5.537	5.803	5.639
p5	-181.3	-175.8	-166.7
R-square	0.9888	0.9817	0.9884
RMSE	1.978	2.533	2.019

Therefore, to generalize (to take all the possible values into account) and for the sake of simplicity, a fit curve has been generated for each channel (i.e., for each curve of the three conversion table curves shown in figure 2).

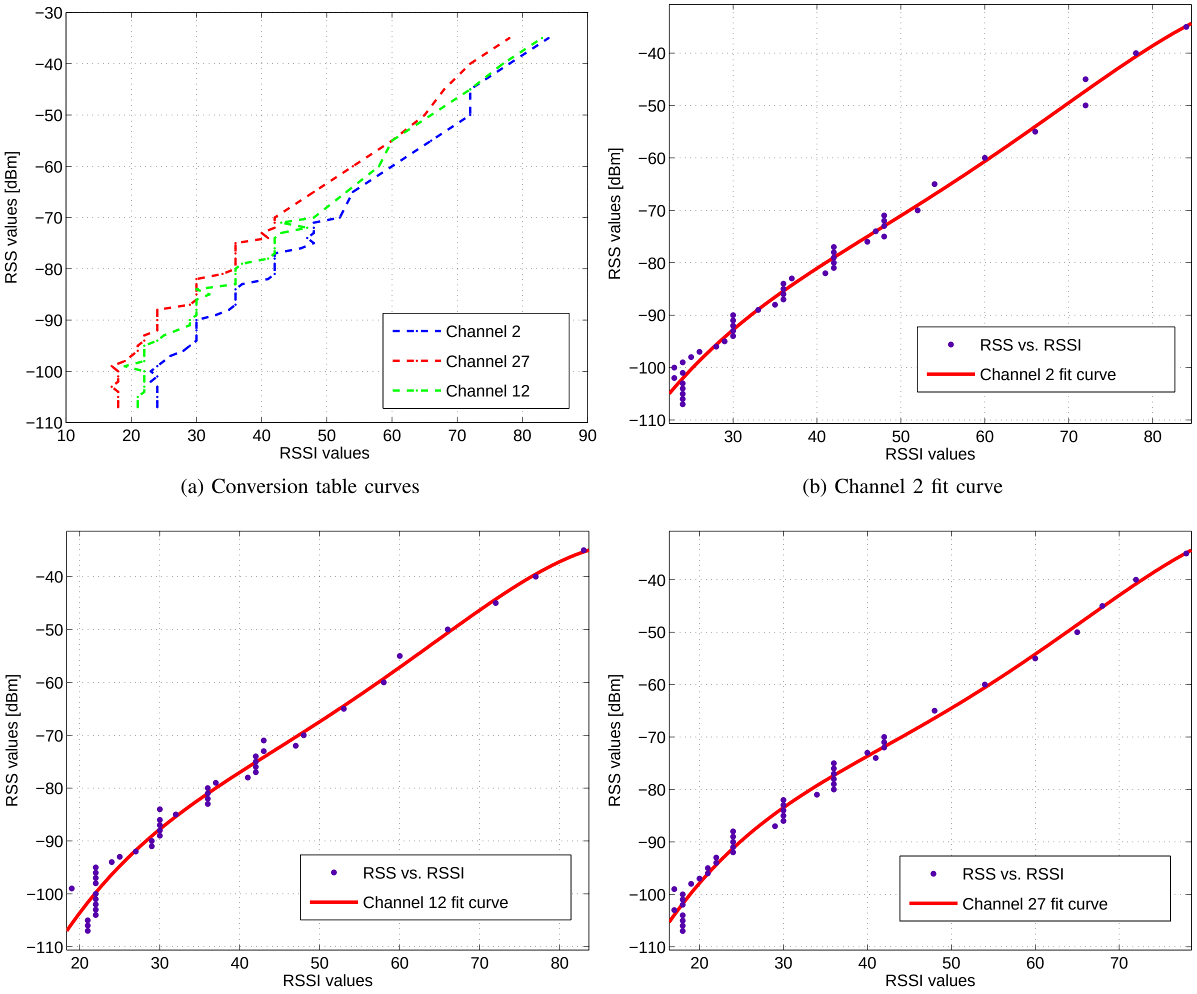
The modem has been calibrated only for three channels: channel 2, channel 12 and channel 27. The channels have been chosen to cover the used frequency range, channel 2 is the lower frequency bound, channel 12 is the middle frequency and channel 27 is the higher frequency bound. If the number of the observed (measured) channel does not exist in the calibration table, the closest channel is to be chosen.

Three curves have been generated depending on the calibration table, one curve for each channel. The fitting equation is given by:

$$\begin{aligned} RSS &= p1 \times RSSI^4 + p2 \times RSSI^3 + p3 \times RSSI^2 \\ &\quad + p4 \times RSSI + p5 \end{aligned} \qquad (2)$$

The Coefficients p1, p2, p3, p4 and p5 differ from one channel to another as shown in table I. Therefore, it is enough to store only the coefficients instead of storing the complete calibration table. *R-square* measures how successful the fit is in explaining the variation of the data; a value closer to 1 indicates a better fit; and RMSE is the root mean squared error, a value closer to 0 indicates a better fit. One can choose a less complicated fitting curve (linear curve), but the 4^{th} degree polynomial curve was chosen, because it gives the highest *R-square* value and the lowest root mean squared error, with a price of only storing few numbers (coefficients) more.

Obtaining RSS values from SCORE ones requires knowing *AvgViterbi* information (refer to equation 1). Unfortunately this information is not available for all the measured BSs (It is only available for the serving BS). The available calibration tables show that the value of *AvgViterbi* takes considerable values only for weak signals, and takes the value of zero (or very small) for strong signals. Thus, setting *AvgViterbi* to zero will affect the most the accuracy of low signals. However, this will make the overall SCORE measurements accuracy lower than RSSI one.



(a) Conversion table curves

(b) Channel 2 fit curve

(c) Channel 12 fit curve

(d) Channel 27 fit curve

Fig. 2: The calibration table fitting curves. These curves are used to convert RSSI to RSS values measured in dBm.

D. *Measurement accuracy*

Measuring RSS values directly using advanced measurement equipments (such as spectrum analyzers) is the most accurate way, it gives the values directly in dBm as real values. These values will be considered as our reference values to evaluate the accuracy of the other types of measurements.

1) RSSI measurements accuracy: The RSSI values are less accurate than RSS values due to the following reasons:

- The RSSI values are integer numbers, i.e., each of the original values is rounded to the closest integer.
- Using the modem calibration table will produce an additional error due to calibration table production and the conversion error.
- The modem is calibrated only for three channels, for the lower, medium and high frequency bands, and the closest channel to the measured one will be used. This approximation will affect the accuracy of the obtained values.
- The fitting curves are also approximations to real values, thus using these fitting curves to convert RSSI to RSS will affect also the accuracy of the final values.

However, the real RSS values (measured directly in dBm by dedicated equipments) are not used in this research; and from now until the end of this paper the term RSS will be used to indicate the RSS values obtained using RSSI measurements.

2) SCORE measurements accuracy: SCORE values are less accurate than RSSI measurements due to the lack of *AvgViterbi* information. Setting this information to any constant value (such as zero) will affect negatively the conversion accuracy of some values and keep the accuracy unchanged for the rest of the values. Thus, some obtained values will keep their accuracy unchanged and some will lose some accuracy due to this assumption. This approximation will produce an additional error in addition to the error resulting from converting RSSI to RSS. Equation 1 shows that the minimum obtained RSSI value from SCORE measurements is 22. But when RSSI values are obtained by direct measurements, values such as 17, 18, 19, 20, 21 can be found, refer to calibration table shown in figure 1 (recall that the provided calibration table in 1 is only a sample and doesn't show all the values and channels).

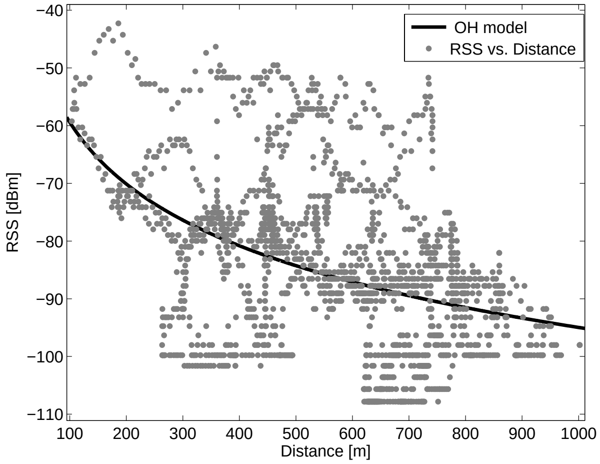


Fig. 3: The actual RSS measurements and the related OH model.

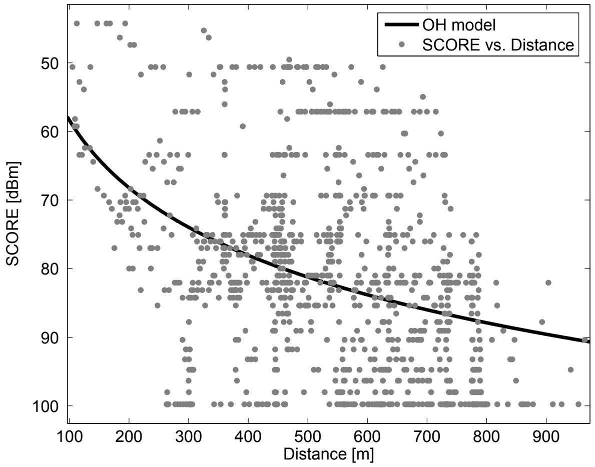


Fig. 4: The actual SCORE measurements and the related OH model.

E. The relation between the measured SCORE and RSS values in the area under study

The relation between the measured SCORE and RSS values in the area under study has been studied by computing the average error and the covariance between the two quantities. The results show good correlation between SCORE and RSS values which means that using SCORE values instead of RSS values is possible, but lower positioning accuracy is expected. Some antennas (BSs) have better correlation between the SCORE and RSS values comparing to other antennas. This is because these antennas have a stronger signal in the area under study. For example antennas 1,2,3 and 4 have better correlation than antennas 6 and 7 (refer to figure 5). Indeed the BSs 6 and 7 are experimental BSs and have lower transmission power than the rest of the BSs in the area under study. Also, the correlation is bad for low signals in all the antennas. This is due to the approximation made about *AvgViterbi* value. This value was set to zero while for low signals *AvgViterbi* takes large values refer to figure 1. Figure 3 and figure 4 depict the actual RSS and SCORE values along with the related OH model. The two obtained OH models show good correlation between the two quantities and proves that using SCORE values to obtain localization is plausible.

In this research, the focus is on using the practically available RSS-based measurements for localization. In fact the only practically available RSS-based measurements for localization is the SCORE measurements. Because they are the only RSS-based measurements that can be obtained simultaneously for all the available BSs, which is indeed a vital condition for realistic applications.

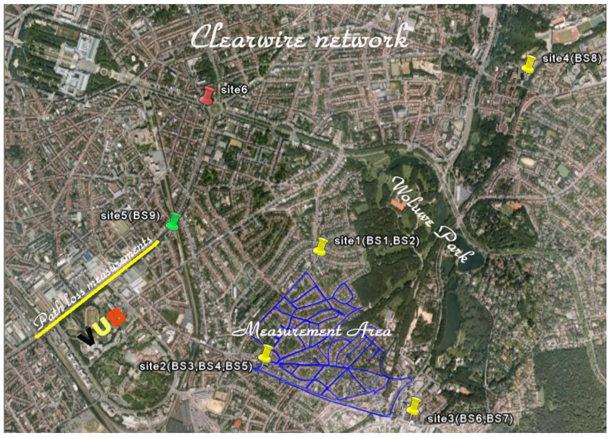


Fig. 5: The area under study, the measurement area is the blue roads.

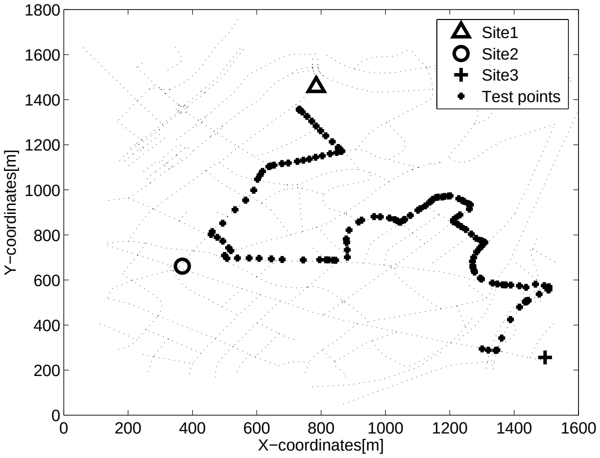


Fig. 6: The used test points. Note that the test points were chosen to cover almost the whole area.

IV. Fingerprinting-based Localization depending on RSS-based measurements

Localization depending on RSS measurements suffers from the high variations in signal strength due to the propagation environment effect on the traveling signal from the transmitter to the receiver. Therefore, the focus is on finding localization approaches that can cope with these variations and minimize their effect on positioning accuracy. one of the most known

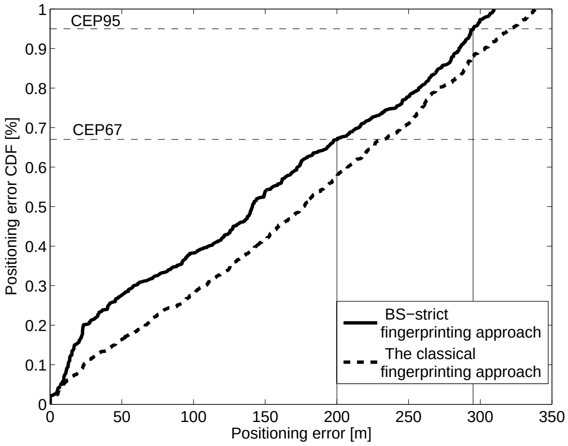


Fig. 7: The positioning error CDFs. The two fingerprinting approaches were used, the classical and the BS-strict.

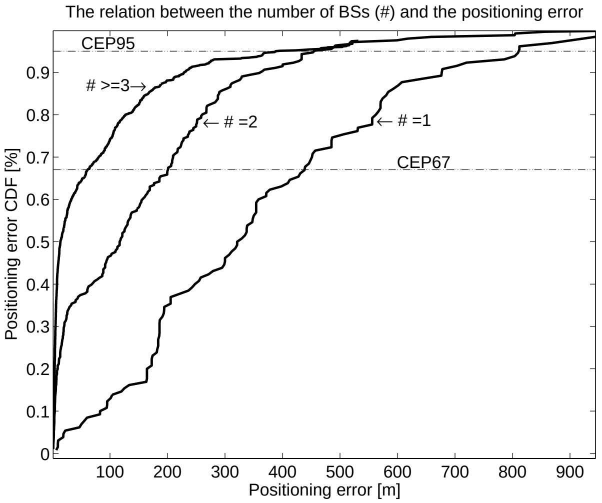


Fig. 8: The relation between the number of the detected BSs and the positioning error.

and effective approach is the *Fingerprinting*.

In general, The received power can be expressed as

$$P_r = \frac{a_t P_t}{R^2} + n_t. \tag{3}$$

Where, P_r is the received power, P_t denotes the transmitted power, a_t is a constant related to the signal frequency and the gain of the receiver and transmitter antennas, R is the distance between the transmitter and the receiver and n_t is the noise component. The previous equation shows that the transmitted power decades with the distance between the transmitter and the receiver. This fact has been used to estimate the distance between the receivers (SSs) and the transmitters (BSs) to obtain localization. The classical model of RSS measurements is based on the so-called Okumura-Hata model [7], [8] which is given as

$$\text{OH model:} \quad y = P_{BS} - 10\alpha \log_{10}(\|p_{BS} - p_{SS}\|_2) + n. \tag{4}$$

where P_{BS} is transmitted signal power (in dB); α is the path loss exponent; n is the measurement noise, p_{BS} is the position of the BS and p_{SS} the position of SS; the standard $\|\cdot\|_2$ norm is used. This model has been used in many proposed localization algorithms [3], [16], but it suffers from the following shortcomings:

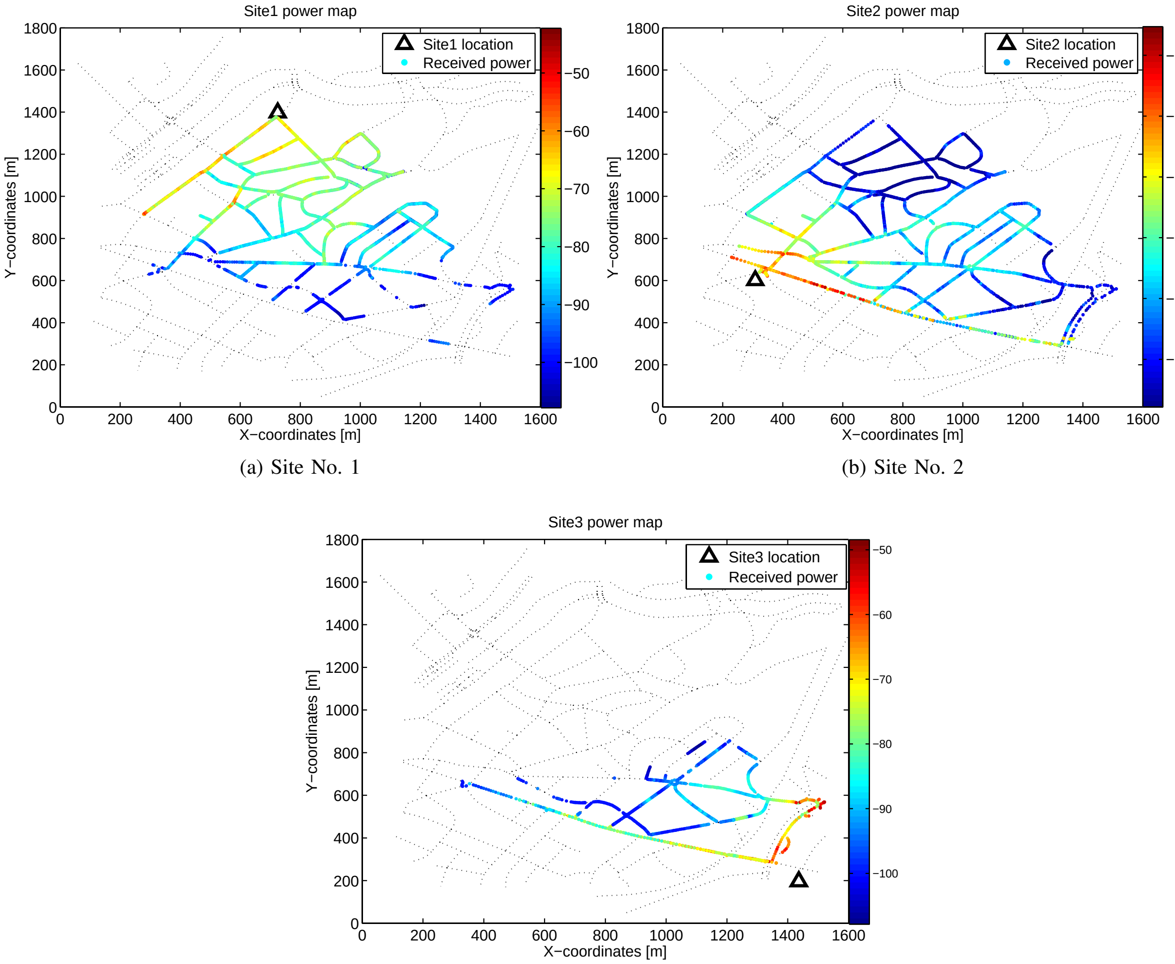
1) It is global, this means that the used path loss exponent is not accurate, because it is related directly to the local environment.
2) The position of the transmitters needs to be known.
3) The transmitted power also needs to be known.

And, the most important is the high signal variations due to the fading, especially is urban environments where the multipath phenomena is strongly present. Figure 3 depicts the actual RSS measurements for one of the base stations in the area under study and the related OH model. It is clear that the difference between the actual measured values and the ones obtained depending on theoretical models (such as OH) is relatively big and plays an opposing role on localization accuracy. Therefore, obtaining localization depending on a certain model will be strongly affected by the difference between the actual measured values and the values obtained by using this model. And the more the model can minimize this difference, the best localization accuracy can be obtained depending on this model such as in fingerprinting localization.

Currently, fingerprinting-based positioning in wireless networks is a new and very active field. The key idea of fingerprinting -as the name says- is that each location has a set of unique features, and this set of features or the “fingerprints” will be used to identify a specific location in the same way a person’s fingerprint is used to identify him / her. To be able to use this methodology a database of all the fingerprints has to be ready and stored on the system as vectors $[(x, y), F_v]$ where, (x, y) is the location coordinates and F_v is the set of the considered features in the said location. The features could be any wireless network related values like: TA, AOA, RSS or a combination of them. This database implicitly takes care of the line of sight (LOS) and non-line of sight (NLOS) problems that are difficult to handle [17]; and partially includes the effects of slow and fast fading. The total effect can be approximated as a gain in SNR with a factor of ten compared to the OH model, see [3]. The mentioned database can be built using two approaches:

1) Collect the “fingerprints” by using direct measurements. This method gives the most accurate database but it is time consuming.
2) Predict the “fingerprints” to avoid the time consuming job by using the radio propagation formulas to predict the RSS values. This method is not accurate as the first one because it is not possible to model all the propagation effects.

In this research, the first method was adopted, and the RSS values (the used feature) have been collected from all the possible roads in the area under study, we assume that the target or the user is using the public road network (in fact this assumption has no value in our case as we obtain localization



(a) Site No. 1

(b) Site No. 2

(c) Site No. 3

Fig. 9: The power maps of the available WiMAX sites in area under study.

in the static case). The RSS values were collected by using a special calibrated modem with extra software installed on (provided by Clearwire, Belgium). The measurements have been manipulated and stored in a database for later use. The area under study is the measurement area shown in figure 5.

The power maps of all the available sites in the measurement area have been generated and plotted, see figure 9. The mentioned power maps were manipulated and stored in a database in forms of vectors (RSS vectors). Each vector contains all the RSS values in a specific location (x, y); i.e. the database contains a set of vectors (fingerprints) of the form $[(x, y), RSS_1, RSS_2, ...RSS_n]$, where n is the number of the received base stations in the location (x, y).

The localization is obtained during the fingerprinting on-line phase by finding the best match between the target's (user's) vector (RSS vector) and the vectors stored in the database or, in other words, compare the target's "fingerprint" with all the database "fingerprints" and choose the best match. The matching process is based on calculating the distance between the target's fingerprint with all the fingerprints stored in the database, and choosing the fingerprint with the minimum distance. The current WiMAX modems don't measure RSS values, but instead they measure *RSS-based* values called SCORE values; therefore, the target's fingerprint will contain the set of the received SCORE values, and the online measurement vector is of the form $[SCORE_1, SCORE_2, ...SCORE_m]$, where m is the number of the received base stations.

A. Fingerprinting-based localization using SCORE measurements

The on-line target's fingerprint contains all the received SCORE values in the target's current location. A distance metric approach was used to compare all the "fingerprints" stored in the database with the target's fingerprint . In case of comparing vectors with different lengths (this happens when some BSs are not received by the target but they already exist in the database's vector or the vise versa, when the database's vector doesn't contain some BSs that have been received by the target); the missing information is considered as not a number value (NaN) and two approaches were followed:

1) *The classical fingerprinting approach*: Ignore the NaN values and compute the distance between the two vectors.

2) *The BS-strict fingerprinting approach*: Hard punishment to be applied on the points that have NaN values by excluding them (the distance between the two vectors is considered to be infinite). It is called *"BS-strict"* because the two vectors must have the same BSs to be considered.

The SCORE values were obtained using the test points shown in figure 6. While choosing the test points, a special attention has been paid to considering all the possible situations; i.e., in some points only one BS can be measured (one SCORE value), in others 2 or more BSs can be measured, etc.... Building the off-line database is the main concern in fingerprinting localization. It requires extensive measurement campaigns, and once it is built, it has to be updated continuously to stay consistent with the changing environment (new BSs, new constructions etc...). More convenient solution than conducting measurements is to use radio planing programs. This solution produces less accurate databases but it is preferred in some cases. Another alternative is to allow users with positioning capabilities (such as GPS) to contribute on-line to the database. Some users could be stationary (at home, in the office), others could be mobile users (for example a taxi driver). This option is a good way to keep the database accuracy and consistency, and to overcome the direct measurements difficulties. In this research, the off-line database was built using direct measurements, and the RSS values were chosen to build the database due to the following reasons:

1) Using radio planning tools produces RSS databases only.
2) Some RSS databases are already available and provided by some specialized companies for all the available wireless networks.
3) SCORE values are subject to higher variations than RSS ones. Because they depend on the signal strength and the output of the *Viterbi* decoder.

The two mentioned fingerprinting approaches were applied, the classical and the BS-strict approach. The obtained results show that the BS-strict approach performance is slightly better than the classical one. This is due to the fact that the BSID values (Cell-IDs values) are more robust against the noise than the RSS-based values; i.e. the same Cell-ID will be read regardless the presence of a strong noise or not, but different RSS-based values will be read.

B. The relation between the positioning accuracy and the number of the detected base stations

The relation between the positioning accuracy and the number of the detected BSs has been studied using RSS values. Four positioning accuracy intervals have been considered as shown in table II. The percentage of the points that have one BS (type 1), two BSs (type 2)...etc, has been calculated for each accuracy interval. It has been found that for high accuracy intervals, most of the points have 3 BSs or more; and for low positioning accuracy, most of the points have only one BS. Figure 8 plots the relation between the number of the detected BSs and the positioning accuracy.

TABLE II: The relation between the positioning accuracy and the number of the detected BSs. The percentage of each type has been calculated in each accuracy interval. The table has to be read column by column.

Number of BSs	1	2	3 (or more)
Positioning error is <10m	17%	34%	49%
Positioning error is from 10 to 50m	8%	19%	18%
Positioning error is from 51 to 350m	38%	37%	31%
Positioning error is > 350m	37%	10%	2%

Table II gives detailed results about the percentage of each type in each positioning accuracy interval. For example, consider type 1 (i.e., the fingerprint vector contains only one BS), we found that only 25% (17% + 8%) of the points were positioned in an accuracy less or equal to 50 m against 75% of the points that were positioned with low accuracy . On the other hand, 67% (49% +18 %) of the points that have 3 or more BSs were positioned with high accuracy ($\leq$ 50 m) against 33% of the points that were positioned with low accuracy. In fact increasing the length of the fingerprints increases their diversity (higher distinction between the points); and therefore, increases the possibility of making the right match between the on-line fingerprint and the stored ones (database). Therefore, the localization accuracy will be improved by increasing the network density as more BSs will be available.

V. Conclusion

This paper has discussed using fingerprinting-based positioning to locate users in WiMAX networks depending on SCORE measurements. A novel fingerprinting approach was introduced depending on the fact that base stations identifications numbers (Cell-IDs) are more robust against the multipath and fading than RSS-based values. The new approach is called BS-strict because the compared vectors (i.e., fingerprints) must have the same BS values to match. The relation between the number of the detected base stations and the positioning accuracy has been investigated and it has been found that increasing the network density will increase the positioning accuracy.

The obtained results show that using SCORE values to position users in WiMAX networks is feasible with enough positioning accuracy to satisfy most of the known LBS requirements. The achieved accuracy is obtained in the static case (static positioning) and the target's motion information was not used. In dynamic positioning, the target's motion information will be used and better positioning accuracy is expected. Moreover, using the public road network information (assuming that the user is always on the road) will produce additional positioning accuracy improvement [18]. The future work will focus on improving the accuracy by using the motion model information in addition to the public road network information.

References

[1] M. Bshara and L. V. Biesen, “Fingerprinting-Based Localization in WiMAX Networks Depending on SCORE Measurements,” *AICT 2009 conference proceedings*, pp. 9–14, May 2009.

[2] Cello consortium report. [Online]. Available: http://www.telecom.ntua.gr/cello/documents/CELLO-WP2-VTT-D03-007-Int.pdf

[3] F. Gustafsson and F. Gunnarsson, “Possibilities and fundamental limitations of positioning using wireless communication networks measurements,” *IEEE Signal Process. Mag.*, vol. 22, pp. 41–53, 2005.

[4] G. Sun, J. Chen, W. Guo, and K. Liu, “Signal processing techniques in network-aided positioning: A survey of state-of-the-art positioning designs,” *IEEE Signal Process. Mag.*, vol. 22, no. 4, pp. 12–23, July 2005.

[5] S. Gezici, Z. Tian, B. Giannakis, H. Kobayashi, and A. Molisch, “Localization via ultra-wideband radios,” *IEEE Signal Process. Mag.*, vol. 22, pp. 70–84, 2005.

[6] D. Li and Y. Hu, “Energy-based collaborative source localization using acoustic microsensor array,” *Journal of Applied Signal Processing*, pp. 321–337, 2003.

[7] Y. Okumura, E. Ohmori, T. Kawano, and K. Fukuda, “Field strength and its variability in VHF and UHF land-mobile radio service,” *Rev. Elec. Commun. Lab.*, vol. 16, pp. 9–10, 1968.

[8] M. Hata, “Empirical formula for propagation loss in land mobile radio services,” *IEEE Trans. Veh. Technol.*, vol. 29, no. 3, pp. 317–325, Aug. 1980.

[9] W. D. Wang and Q. X. Zhu, “RSS-based Monte Carlo localisation for mobile sensor networks,” *IET Communications*, pp. 673–681, 2008.

[10] M. Nabaee, A. Pooyafard, and A. Olfat, “Enhanced object tracking with received signal strength using Kalman filter in sensor networks,” *Internatioal Symposium on Telecommunications*, pp. 318–323, 2008.

[11] J. Shirahama and T. Ohtsuki, “RSS-based localization in environments with different path loss exponent for each link,” *Vehicular Technology Conference*, pp. 1509–1513, 2008.

[12] M. Zhang, S. Zhang, J. Cao, and H. Mei, “A novel indoor localization method based on received signal strength using discrete Fourier transform,” *Communications and Networking in China*, pp. 1–5, 2006.

[13] O. Sallent, R. Agusi, and X. Cavlo, “A mobile location service demonstrator based on power measurements,” *Vehicular Technology Conference*, vol. 6, no. 1, pp. 4096–4099, Sep. 2004.

[14] A. Taok, N. Kandil, S. Affes, and S. Georges, “Fingerprinting localization using ultra-wideband and neural networks,” *Signals, Systems and Electronics*, vol. 54, no. 4, pp. 529–532, Aug. 2007.

[15] *Local and Metropolitan Area Networks Part 16: Air Interface for Fixed Broadband wireless Access Systems*, IEEE Std. 802.16, 2004.

[16] A. Heinrich, M. Majdoub, J. Steuer, and K. Jobmann, “Real-time path-loss position estimation in cellular networks,” in *Proceedings of International Conference on Wireless Networks (ICWN’02)*, Jun. 2002.

[17] K.-T. Feng, C.-L. Chen, and C.-H. Chen, “GALE: an enhanced geometry-assisted location estimation algorithm for NLOS environments,” *IEEE Trans. Mobile Comput.*, vol. 7, no. 2, pp. 199–213, Feb. 2008.

[18] U. Orguner, T. B. Schön, and F. Gustafsson, “Improved target tracking with road network information,” *Proceedings of IEEE Aerospace Conference*, March 2009.

K-Means on the Graphics Processor: Design And Experimental Analysis

Mario Zechner
Know-Center
Inffeldgasse 21a
Graz, Austria
mzechner@know-center.at

Michael Granitzer
Graz Technical University
Inffeldgasse 21a
Graz, Austria
mgranitzer@tugraz.at

Abstract—**Apart from algorithmic improvements many intensive machine learning algorithms can gain performance by parallelization. Programmable graphics processing units (GPU) offer a highly data parallel architecture that is suitable for many computational tasks in machine learning. We present an optimized k-means implementation on the graphics processing unit. NVIDIA's Compute Unified Device Architecture (CUDA), available from the G80 GPU family onwards, is used as the programming environment. Emphasis is placed on optimizations directly targeted at this architecture to best exploit the computational capabilities available. Additionally drawbacks and limitations of previous related work, e.g. maximum instance, dimension and centroid count are addressed. The algorithm is realized in a hybrid manner, parallelizing distance calculations on the GPU while sequentially updating cluster centroids on the CPU based on the results from the GPU calculations. An empirical performance study on synthetic data is given, demonstrating a maximum 14x speed increase to a fully SIMD optimized CPU implementation. We present detailed empirical data on the runtime behavior of the various stages of the implementation, identify bottlenecks and investigate potential discrepancies arising from different rounding modes on the GPU and CPU based. We extend our previous work in [1] by giving a more in depth description of CUDA as well as including previously omitted experimental data.**

Keywords-**Parallelization, GPGPU, K-Means**

I. Introduction

In the last decades the immense growth of data has become a driving force to develop scalable data mining methods. Machine learning algorithms have been adapted to better cope with the mass of data being processed. Various optimization techniques lead to improvements in performance and scalability among which parallelization is one valuable option.

One of the many data mining methods widely in use is partitional clustering which is formally defined as "the organization of a collection of patterns (usually represented as a vector of measurements, or a point in a multidimensional space) into clusters based on similarity" [2]. The application of clustering is widespread among many different fields, such as computer vision [3], computational biology [4, 5] or text mining [6]. A non-optimal solution to the NP-hard problem of partitional clustering was proposed by Lloyd in [7]. The most well known variant is the k-means algorithm in [8]. The popularity of k-means is explainable by its low computational complexity and well understood mathematical properties. However, k-means will only find non-optimal local-minima, depending on the initial configuration of centroids. This is also known as the seeding problem and was addressed in various works. Recently a new strategy yielding better clustering results was introduced in [9]. Still, the runtime performance of k-means is a concern as data is growing rapidly, especially when finding the correct parameter of k can only be done by performing several runs with different numbers of clusters and initial seedings.

With the appearance of programmable graphics hardware in 2001, using the GPU as a low-cost highly parallel streaming co-processor became a valuable option. Figure (I) illustrates the performance of GPUs and CPUs as well as differences in memory throughput over the years 2003 to 2008. This developement spawned scientific interest in this new architecture and resulted in numerous publications demonstrating the advantages of GPUs over CPUs when used for data parallel tasks. Much attention was focused on transferring common parallel processing primitives to the GPU and creating frameworks to allow for more general purpose programming [10, 11]. The most problematic aspect of this undertaking was transforming the problems at hand into a graphics processor pipeline friendly format, a task needing knowledge about graphics programming. The reader is referred to [12] where an in-depth discussion on mapping computational concepts to the GPU can be found. This entry barrier was recently lowered by the introduction of NVIDIA's CUDA [13] as well as ATI's Close to Metal Initiative [14]. Both were designed to enable direct exploitation of the hardware's capabilities circumnavigating the invocation of the graphics pipeline via an API such as OpenGL or DirectX. In this work CUDA was chosen due to its more favorable properties, namely the high-level approach employed by its seamless integration with C and the quality of its documentation.

In this paper a parallel implementation of k-means on the GPU via CUDA is discussed. Section II discusses the sequential and parallel variants of k-means leading to Section III where related work is investigated. Section IV gives an overview of CUDA's properties and programming

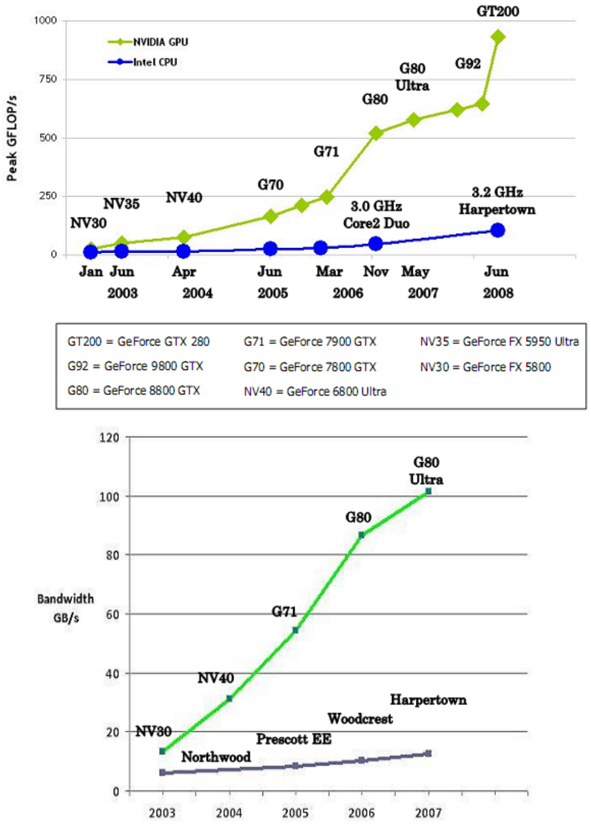


Figure 1. Evolution of CPU and GPU performance over the last decade [15]

model followed by Section V describing the concrete parallel implementation of k-means on the GPU. A comparison of the GPU implementation versus an optimized sequential CPU implementation is given in Section VI. Finally, Section VII concludes this paper.

II. K-Means Clustering

In this section a definition of the k-means problem is given as well as non-optimal sequential and parallel algorithmic solutions. Additionally the computational complexity is discussed.

A. Problem Definition

The k-means problem can be defined as follows: a set $\mathcal{X}$ of n data points $\mathbf{x_i} \in \mathbb{R}^d, i = 1, \dots, n$ as well as the number of clusters $k \in \mathbb{N}^+ < n$ is given. A cluster $\mathcal{C}_j \subset \mathcal{X}, j = 1, \dots, k$ with a centroid $\mathbf{c_j} \in \mathbb{R}^d$ is composed of all points in $\mathcal{X}$ for which $\mathbf{c_j}$ is the nearest centroid. The distance from a data point to a centroid is determined by some suitable metric. Figure (II-A) shows a simple toy dataset with 3 gaussian clusters, some outliers and the 3 cluster centroids. The optimal set $\mathcal{C}$ of k centroids can be found by minimizing the following potential function:

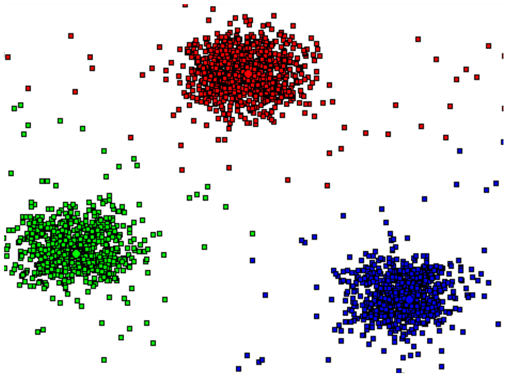


Figure 2. A simple two dimensional toy example with three gaussian clusters.

$$\phi = \sum_{i=1}^{n} \min_{\mathbf{c_j} \in \mathcal{C}} \mathcal{D}(\mathbf{x_i}, \mathbf{c_j})^2 \qquad (1)$$

$\mathcal{D}$ is a metric in $\mathbb{R}^d$, usually the euclidean distance. Solving Equation (1) even for two clusters was proven to be NP-hard in [16]. However, a non-optimal solution for the k-means problem exists and will be described in the following subsection. For the rest of the discussion it is assumed that the set of data points $\mathcal{X}$ is already available in-core, that is loaded to memory.

B. Sequential K-Means

In [8] MacQueen describes an algorithm that locally improves some clustering $\mathcal{C}$ by iteratively refining it. An initial clustering $\mathcal{C}$ is created by choosing k random centroids from the set of data points $\mathcal{X}$. This is known as the seeding stage. Next a labeling stage is executed where each data point $\mathbf{x_i} \in \mathcal{X}$ is assigned to the cluster C_j for which $\mathcal{D}(\mathbf{x_i}, \mathbf{c_j})$ is minimal. Each centroid $\mathbf{c_j}$ is then recalculated by the mean of all data points $\mathbf{x_i} \in \mathcal{C}_j$ via $\mathbf{c_j} = \frac{1}{|\mathcal{C}_j|} \sum_{\mathbf{x_i} \in \mathcal{C}_j} \mathbf{x_i}$. The labeling and centroid update stage are executed repeatedly until $\mathcal{C}$ no longer changes. This procedure is known to converge to a local minimum subject to the initial seeding [17]. Algorithm 1 describes the procedure in algorithmic terms.

The next subsection demonstrates how this sequential algorithm can be transformed into a parallel implementation.

C. Parallel K-Means

In [18] Dhillon presents a parallel implementation of k-means on distributed memory multiprocessors. The labeling stage is identified as being inherently data parallel. The set of data points $\mathcal{X}$ is split up equally among p processors, each calculating the labels of all data points of their subset

Algorithm 1 Sequential K-Means Algorithm

```
$\mathbf{c_j} \leftarrow$ random $\mathbf{x_i} \in \mathcal{X}, j = 1, \ldots, k,$ s.t. $\mathbf{c_j} \neq \mathbf{c_i} \forall i \neq j$
repeat
  $\mathcal{C}_j \leftarrow \emptyset, j = 1, \ldots, k$
  for all $\mathbf{x_i} \in \mathcal{X}$ do
    $j \leftarrow \arg\min \mathcal{D}(c_j, x_i)$
    $\mathcal{C}_j \leftarrow \mathcal{C}_j \cup x_i$
  end for
  for all $\mathbf{c_j} \in \mathcal{C}$ do
    $\mathbf{c_j} \leftarrow \frac{1}{|\mathcal{C}_j|} \sum_{\mathbf{x_i} \in \mathcal{C}_j} \mathbf{x_i}$
  end for
until convergence
```

of $\mathcal{X}$. In a reduction step the centroids are then updated accordingly. It has been shown that the relative speedup compared to a sequential implementation of k-means increases nearly linearly with the number of processors. Performance penalties introduced by communication cost between the processors in the reduction step can be neglected for large n.

Algorithm 2 Parallel K-Means Algorithm

```
if threadId = 0 then
  $\mathbf{c_j} \leftarrow$ random $\mathbf{x_i} \in \mathcal{X}, j = 1, \ldots, k,$ s.t. $\mathbf{c_j} \neq \mathbf{c_i} \forall i \neq j$
end if
synchronize threads
repeat
  for all $\mathbf{x_i} \in \mathcal{X}_{threadId}$ do
    $l_i \leftarrow \arg\min \mathcal{D}(c_j, x_i)$
  end for
  synchronize threads
  if threadId=0 then
    for all $\mathbf{x_i} \in \mathcal{X}$ do
      $\mathbf{c_{l_i}} \leftarrow \mathbf{c_{l_i}} + \mathbf{x_i}$
      $m_{l_i} \leftarrow m_{l_i} + 1$
    end for
    for all $\mathbf{c_j} \in \mathcal{C}$ do
      $\mathbf{c_j} \leftarrow \frac{1}{m_j} \mathbf{c_j}$
    end for
    if convergence then
      signal threads to terminate
    end if
  end if
until convergence
```

Since the GPU is a shared memory multiprocessor architecture this section briefly outlines a parallel implementation on such a machine. It only slightly diverges from the approach proposed by Dhillon. Processors are now called threads and a master-slave model is employed. Each thread is assigned an identifier between 0 and $t-1$ where t denotes the number of threads. Thread 0 is considered the master thread, all other threads are slaves. Threads share some memory within which the set of data points $\mathcal{X}$, the set of current centroids $\mathcal{C}$ as well as the clusters $\mathcal{C}_j$ reside. Each thread additionally owns local memory for miscellaneous data. It is further assumed that locking mechanisms for concurrent memory access are available. Given this setup the sequential algorithm can be mapped to this programming model as follows.

The master thread initializes the centroids as it is done in the sequential version of k-means. Next $\mathcal{X}$ is partitioned into subsets $\mathcal{X}_i, i = 0, \ldots t$. This is merely an offset and range calculation each thread executes giving those $\mathbf{x_i}$ each thread processes in the labeling stage. All threads execute the labeling stage for their partition of X. The label of each data point $\mathbf{x_i}$ is stored in a component l_i of an n-dimensional vector. This eliminates concurrent writes when updating clusters and simplifies bookkeeping. After the labeling stage the threads are synchronized to ensure that all data for the centroid update stage is available. The centroid update stage could then be executed by a reduction operation. However, for the sake of simplicity it is assumed that the master thread executes this stage sequentially. Instead of iterating over all centroids the master thread iterates over all labels partially calculating the new centroids. A k-dimensional vector $\mathbf{m}$ is updated in each iteration where each component m_j holds the number of data points assigned to cluster $\mathcal{C}_j$. Next another loop over all centroids is performed scaling each centroid $\mathbf{c_j}$ by $\frac{1}{m_j}$ giving the final centroids. Convergence is also determined by the master thread by checking whether the last labeling stage introduced any changes in the clustering. Slave threads are signaled to stop execution by the master thread as soon as convergence is achieved. Algorithm 2 describes the procedure executed by each thread.

D. Computational Complexity

In this section the number of operations executed by k-means in each iteration is investigated. This number is equal for both implementations. It therefore serves as the basis for comparing runtime behavior in section VI.

For the computational complexity analysis each floating point operation is counted as one computational unit. Additions, multiplications and comparisons are considered to be floating point operations. Also, the seeding stage is ignored in this analysis.

The labeling stage consists of evaluating the distance from each data point $\mathbf{x_i}$ to each centroid $\mathbf{c_j}$. Given an euclidean distance metric each distance calculation consists of one subtraction, one multiplication and one addition per dimension totaling in $3d$ operations. Additionally a square root is calculated adding another operation per distance calculation. Finding the centroid nearest to a data point $\mathbf{x_i}$ is an iterative process where in each iteration a comparison between the last minimal distance and the current distance is performed. This adds another operation to the total number of operations per labeling step. There is a total of nk labeling steps resulting in the total numbers of operations of

$$O_{labeling} = 3nkd + 2nk = nk(3d + 2) \tag{2}$$

for the labeling stage in each iteration.

In each iteration of the centroid update stage the mean for each cluster $\mathcal{C}_j$ is calculated consisting of adding $|C_j|$ d-dimensional vectors as well as dividing each component of the resulting vector by $|C_j|$. In total n d-dimensional vectors are added yielding nd operations plus kd operations for the scaling of each centroid $\mathbf{c_j}$ by $\frac{1}{|C_j|}$. For the update stage there are thus

$$O_{update} = nd + kd = d(n + k) \tag{3}$$

operations executed per k-means iteration. The total number of operations per k-means iteration is given by

$$O_{iteration} = O_{labeling} + O_{update} = nk(3d + 2) + d(n + k) \tag{4}$$

From Equations (2) and (3) it can be observed that the labeling stage is clearly the most costly stage per iteration. If $d \ll n$ and $k \ll n$ the updating stage contributes insignificantly to the total number of operations making the labeling stage the dominant factor.

III. Related Work

To the best of the authors' knowledge, three different implementations of k-means on the GPU exist. All three implementations are similar to the parallel k-means implementation outlined in section II-C formulated as a graphics programming problem. Tabel (I) gives an overview of the various approaches.

In [19] Takizawa and Kobayashi try to overcome the limitations imposed by the maximum texture size by splitting the data set and distributing it to several systems each housing a GPU. A solution to this problem via a multi-pass mechanism was not considered. Also the limitation on the maximum number of dimensions was not tackled. It is also not stated whether the GPU implementation produces the same results as the CPU implementation in terms of precision.

Hall and Hart propose two theoretical options for solving the problem of limited instance counts and dimensionality: multi-pass labeling and a different data layout within the texture [20]. None of the approaches have been implemented though. In addition to the naive k-means implementation the data is reordered to minimize the number of distance calculations by only calculating the metrics to the nearest centroids. This is achieved by finding those centroids by traversing a previously constructed kd-tree. The authors could not observe any problems caused by the non standard compliant floating point arithmetic implementations on the GPU, stating that the exact same clusterings have been found.

The approach of Cao et. al. in [21] differs in that the centroid indices are stored in an 8-bit stencil buffer instead of the frame buffer limiting the number of total centroids to 256. Limitations in dimensionality and instance counts due to maximum texture sizes are solved via a costly multi-pass approach. No statements concerning precision of the GPU version were made.

Summarizing the presented previous work the following can be observed:

- All implementations suffer from architectural constraints such as maximum texture size limiting the number of instances, dimensions and clusters. The limitations can only be overcome by employing more costly multi-pass approaches.
- Not all publications state the exact conditions the implementations were tested under. A direct comparison is not strictly possible. However, the given numbers indicate congruent results yielding an average speedup of a factor between 3 and 4.
- The GPU implementation's performance increases as the problem at hand grows bigger in dimensionality as well as instance and centroid count.
- Only one paper mentioned potential impact of the non standard-compliant floating point arithemtics implemented on GPU's. No effects have been observed.

Based on the previous work the main contributions of this paper are as follows:

1) A parallel implementation of standard k-means on NVIDIA's G80 GPU generation using the non-graphics oriented programming model of CUDA.
2) Removal of the limitations inherent to classical graphics-based general purpose GPU programming approaches for k-means, namely the number of instances, dimensions and centroids enabling large scale clustering problems to be tackled on the GPU.
3) Investigation of precision issues due to the non IEEE single precision floating point compliance of modern GPU's.
4) Performance evaluation of the presented implementation in comparison to an aggressively optimized single core CPU implementation, using SSE3 vectorization as well as loop unrolling optimizations, showing high speedups when compared to the average speedup of previous GPU-based implementations.
5) Evaluation of on-chip memory throughput as well as floating point operation performance.

IV. CUDA

With the advent of the unified shader model the separation of vertex and fragment shader processors in hardware

	Takizawa and Kobayashi [19]	Hall and Hart [20]	Cao et. al. [21]
CPU	Intel P4 3.2 Ghz	AMD Athlon 2800+	Intel P4 3.4 Ghz
Compiler	GNU C++ 3.3.5	?	Intel C++
Optimizations	SSE2	?	SSE2, Hyper-Threading
GPU	NVIDIA Geforce 6600 Ultra	NVIDIA GeforceFX 5900	NVIDIA Geforce 6800 GT
Speedup	4	2-3	4

Table I
SUMMARY OF PREVIOUS GPU-BASED K-MEANS IMPLEMENTATIONS. THE COLUMN SPEEDUP GIVES THE RELATIVE SPEEDUP OF THE GPU VERSION TO THE CPU VERSION BASED ON TOTAL RUNTIME

has vanished. Shader processors can now be configured to perform both tasks depending on the requirements of the application [22]. Starting from the G80 family of GPUs NVIDIA supports this new shader model resulting in a departure from previous GPU designs. The GPU is now composed of so called multiprocessors that house a number of streaming processors ideally suited for massively data-parallel computations.

NVIDIA's CUDA is build on top of this new architecture eliminating the need to reformulate computations to the graphics pipeline. The GPU is viewed as a set of multiprocessors executing concurrent threads in parallel. Threads are grouped into thread blocks and execute the same instruction on different data in parallel. One or more thread blocks are directly mapped to a hardware multiprocessor where time sharing governs the execution order. Within one block threads can be synchronized at any execution point. A certain execution order of threads within a block is not guaranteed. Blocks are further grouped into a grid, communication and synchronization among blocks is not possible; execution order of blocks within a grid is undefined. Threads and blocks can be organized in three and two dimensions respectively. A thread is assigned an id depending on its position in the block. A block is also given an id depending on its position within a grid. Figure (IV shows a two dimensional grid of 2 by 3 thread blocks. Each thread block is composed of 3 by 4 threads. The thread and block id of a thread is accessible at runtime allowing for specific memory access patterns based on the chosen layouts. Each thread on the GPU executes the same procedure known as a kernel [15].

Threads have access to various kinds of memory. Each thread has very fast thread local registers and local memory assigned to it. Within one block all threads have access to block local shared memory that can be accessed as fast as registers depending on the access patterns. Registers, local memory and shared memory are limited resources. Portions of device memory can be used as texture or constant memory which benefit from on-chip caching. Constant memory is optimized for read-only operations, texture memory for specific access patterns. Threads also have access to uncached general purpose device memory or global memory [15]. Figure (IV) gives an overview of this architecture.

Various pitfalls exist that can degrade performance of the GPU. Shared memory access by multiple threads in parallel can produce so called bank conflicts serializing execution of those threads and therefore reducing parallelism. Second, when accessing global memory addresses have to be a multiple of 4, 8 or 16, otherwise an access might be compiled to multiple instructions and therefore accesses. Also, addresses accessed simultaneously by multiple threads in global memory should be arranged so that memory access can be coalesced into a single continuous aligned memory access. This is often referred to as memory coalescing. Another factor is so called occupancy. Occupancy defines how many blocks and therefore threads are actually running in parallel. As shared memory and registers are limited resources the GPU can only run a specific number of blocks in parallel. It is therefore mandatory to optimize the usage of shared memory and registers to allow to run as many blocks and threads in parallel as possible [15].

The CUDA SDK gives the developer easy to use tools that fully integrate with various C++ compilers. Code for the GPU is written in a subset of C with some extensions

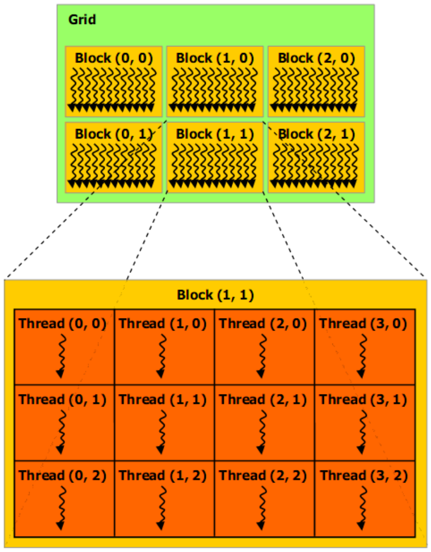


Figure 3. A grid of thread blocks. Each thread block is composed of a number of threads. Blocks and threads are indexed [15]

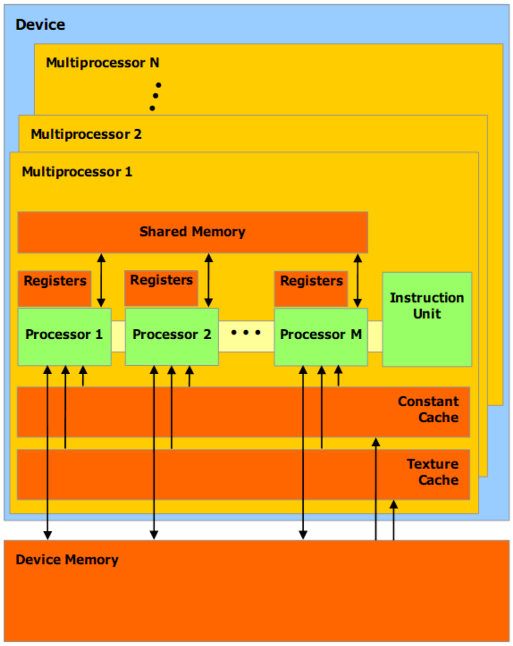


Figure 4. Hardware view of the CUDA architecture showing interdependancies and various forms of memory [15]

and can coexist with CPU (host) code in the same source file. The host code is responsible for setting up the layout of blocks and threads as well as uploading data to the GPU. Kernel execution is performed asynchronously, primitives to synchronize between CPU and GPU code are available. Debugging of device code is possible but only in an emulation environment that runs the kernel on the CPU in heavyweight threads which does not simulate all peculiarities of the GPU. Below a simple example for substracting two vectors is given:

```
__global__ void sub(float* a,
                    float* b,
                    float* out)
{
    int i = threadIdx.x;
    out[i] = a[i] - b[i];
}

int main()
{
    // initialize cuda
    ...

    // allocate and fill memory
    ...

    // execute the kernel
    sub<<<1, d>>>(a, b, c);

    // fetch the result from the gpu
    ..
}
```

A CUDA program most often follows this sequence of steps. First the data on which the computations should be caried out is fetched from a source, e.g. a file on disk and loaded into RAM. Next the CUDA API is used to allocate memory on the GPU and the data is transfered to this newly allocated space. The API returns pointers that can be used as arguments for a kernel invocation later on. At kernel invocation time the number of thread blocks and blocks within a thread is specified and any arguments are passed in. In the above example each thread in the block will calculate the difference of one dimension of the two vectors and place the result in the corresponding element in array out. The index is derived from the threads id given by threadIdx.x. After the kernel has finished the result, for which memory on the GPU was also allocated previously, is transfered back to RAM. Of course this simple example leaves out a lot of features and capabilities offered by the API and special purpose language extensions. For an excellent introduction to CUDA the reader is refered to [23].

The integration with C is seamless, functions marked with the global modifier will be compiled by the NVIDIA nvcc compiler, yielding binary code executable by the GPU. Kernel invocations will also be replaced with code that loads the binary code to the GPU and invokes some driver functions to execute it.

V. Parallel K-Means via CUDA

This section describes the CUDA based implementation of the algorithm outlined in section II-C. In the first sub section the overall program flow is described. The next subsection presents the labeling stage on the GPU followed by section V-C outlining the data layout used and CUDA specific optimizations employed to further speed up the implementation.

A. Program Flow

The CPU takes the role of the master thread as described in section II-C. As a first step it prepares the data points and uploads them to the GPU. As the data points do not change over the course of the algorithm they are only transfered once. The CPU then enters the iterative process of labeling the data points as well as updating the centroids. Each iteration starts by uploading the current centroids to the GPU. Next the GPU performs the labeling as described in section V-B. The results from the labeling stage, namely the membership of each data point to a cluster in form of an index, are transfered back to the CPU. Finally the CPU calculates the new centroid of each cluster based on these labels and performs a convergence check. Convergence is achieved in case no label has changed compared to the last iteration. Optionally a thresholded difference check of the overall movement of the centroids can be performed to avoid iterating infinitely for some special cluster configurations.

B. Labeling Stage

The goal of the labeling stage is to calculate the nearest centroid for each data point and store the index of this centroid for further processing by the centroid update stage on the CPU. Therefore each thread has to calculate which data points it should process, label it with the index of the closest centroid and repeat this for any of its remaining data points. The task for each thread is thus divided into two parts: calculate and iterate over all data points belong to the thread according to a partitioning schema and performing the actual labeling for the current data point. The following paragraphs will thus first discuss the partitioning schema and the first part of this task followed by a description of the actual labeling step.

As discussed in section IV the GPU slightly differs from the architecture assumed in section II-C. Threads are additionally grouped into blocks that share local memory. Instead of assigning each thread a chunk of data points, each block of threads is responsible for one or more chunks. One such chunk contains t data points where t is the number of threads per block. As the amount of threads per block as well as blocks is limited by various factors, such as used registers, each block processes not only one but several chunks depending on the total amount of data points. Denoting the amount of data points by n then

$$n_{chunks} = \lceil n/t \rceil \tag{5}$$

gives the number of chunks to be processed. Note that the last chunk does not have to be fully filled as n does not have to be a multiple of t. This chunks have to be partitioned among the number of blocks b. Two situations can arise:

1) $n_{chunks} \mod b = 0$, no block is idle
2) $n_{chunks} \mod b \neq 0$, $b - n_{chunks}$ blocks are idle

Therefore each block processes at least $\lfloor n_{chunks}/b \rfloor$ chunks. The first $n_{chunks} \mod b$ blocks process the remaining chunks. For each chunk one thread within a block labels exactly one data point. For chunks that have less data points than there are threads within a block some threads will be idle and not process a data point. Based on the partitioning schema described each thread processes at most n_{chunks} data points. For each data point a thread therefore has to calculate the index of the data point based on it's block and thread id. This is done iteratively in a loop. The thread starts by calculating the index of its data point of the first chunk to be processed by the thread's block expressed by $block.id + thread.id$. In each iteration the next data point's index is calculating by adding tb to the last data points index. In case the calculated index is bigger than $n - 1$ the thread has processed all it's data points. No thread can terminate before the other threads within the same block so any thread that is done processing all its data points has to wait for the other threads to finish processing their remaining data points. Therefore each thread iterates $\lceil n/tb \rceil$ times and simply does not execute the labeling code in case its current data point index is bigger than $n-1$. To minimize the number of idling threads it is therefore mandatory to adjust the number of blocks to the number of data points minimizing $n \mod tb$.

The actual labeling stage is again composed of two distinct parts. A thread has to calculate the distance of its current data point to each centroid. In the implementation presented here all threads within a block calculate the distance to the same centroid at any one time. This allows loading the current centroid to the block's local shared memory accessible by all threads within the block. For each centroid the threads within the block therefore each load a component of the current centroid to shared memory. Each thread then calculates the distance from their data point to the centroid in shared memory fetching the data point's components from global memory in a coalesced manner. See section V-C on the data layout used for coalescing reads and writes. Loading the complete centroid to memory limits the amount of dimensions as shared memory is restricted to some value, on the hardware used it's 16 kilobytes. Given that components are encoded as 32-bit floating point values this roughly equals a maximum dimension count of 4000. To allow for unlimited dimensions the process of loading and calculating the distance from a data point to a centroid is done in portions. In each iteration t components of the centroid are loaded to shared memory. For each component the partial euclidean distance is calculated. Depending on d not all threads have to take part in loading the current components to memory, so some threads might idle. When all threads have evaluated the nearest centroid the resulting label, being the index of the centroid a data point is nearest to, is written back to global memory. The labels for each data point are stored in an additional vector component.

After all blocks have finished processing their chunks the CPU is taking over control again, downloading the labels calculated for constructing the new centroids and checking for convergence. The next section describes the data layout as well as other optimizations.

C. Data Layouts & Optimizations

A GPU-based implementation of an algorithm that is memory bound, as is the case with k-means, can yield very poor performance when the GPU's specifics are not taken into account. For memory throughput these specifics depend on the memory type used for storing and accessing data on the GPU as described in section IV. For the k-means implementation presented in this paper global memory was chosen as the storage area for the data points and centroids. As data points are only read during the labeling stage on the GPU, storage in constant or texture memory might have increased memory throughput to some degree. However, texture and constant memory restrict the maximum amount of data and therefore processable data points and centroids, a drawback earlier GPU-based k-means implementations suffered from as described in section III. Global memory on the other hand allows gather and scatter operations and permits to use almost all of the memory available on the GPU. For global memory coalescing reads and writes are mandatory to achieve the best memory throughput. All vectors are assumed to be of dimensionality d and stored in dense form.

As described in the last section centroids are loaded from global memory to shared memory in portions, each portion being made up of at most t components. As t threads read in subsequent components at once the centroids are stored as rows in a matrix, achieving memory coalescing.

Data points are stored differently due to the order in which components are accessed. Here, each thread accesses one component of its current data point simultaneously to the other threads. Therefore data points are stored column wise again providing memory coalescing. Additionally a component is added as the first component of each vector where each threads writes the label of the closes centroid to for further processing by the CPU. This layout also allows downloading this labels in a bulk operation.

For both centroids and data points special CUDA API methods where used that allocate memory at address being a multiple of 4 yielding the best performance.

As the implementation of k-means using an euclidean distance metric is clearly memory bound further optimizations have been made by increasing occupancy. This was achieved by decreasing the amount of registers each thread uses. Specifically counter variables for the outer loops are stored in shared memory. This optimization increased performance by around 25%. The program executed by each thread uses 10 registers. The optimal number of threads is therefore 128 according to the NVIDIA CUDA Occupancy Calculator included in the CUDA SDK.

As descibed in section V-B partial or entire thread blocks can be idle depending on the ratio between the number of blocks and threads within a block to the number of data points. To reduce the effect of idle blocks on performance the block count is adapted to the number of data points to be processed, minimizing $n_{chunks} \mod b$.

The next section discusses experiments and their results for the k-means implementation presented in this section.

VI. Experiments & Results

Experimental results were obtained on artificial data sets. As the performance is not dependent on the actual data distribution the synthetic data sets were composed of randomly placed data points. To observe the influence of the number of data points on the performance data sets with 500, 5000, 50,000 and 500,000 instances were created. For each instance count 3 data sets were created with 2, 20 and 200 dimensions.

The sequential k-means implementation and the centroid update phase for the gpu-base k-means was coded in C using the Visual C++ 2005 compiler as well as the Intel C++ compiler 10.1. For both compilers full optimizations were enabled, favoring speed over size as well as using processor specific extensions like SSE3. In the case of the Intel C++ compiler all vector related operations such as distance measurements, additions and scaling were vectorized using SSE3. The CUDA portions of the code were compiled using the CUDA Toolkit 2.0.

We evaluated our implementation on two systems. The first one which we will refer to as System 1 was composed of an Intel Core 2 Duo E8400 CPU, 4 GB RAM running Windows XP Professional with Service Pack 3. The GPU was an NVIDIA GeForce 9600 GT hosting 512 MB of RAM, the driver used was the NVIDIA driver for Windows XP with CUDA support version 178.08. The second system, refered to as System 2 was made of an Intel Core 2 Quad Q9550, 4 GB RAM running Windows 7 RC build 7100 and the NVIDIA driver for Windows 7 with CUDA support version 190.39. We first discuss the findings on System 1 and then briefly analyse the results from System 2.

Figures 5 and 6 present the speedups gained by using the GPU relative to the CPU implementation on System 1. While full optimizations were turned on for the Visual C++ version the GPU-based implementation outperformed it by a factor of 4 to 43 for all but the smallest data set. A clear increase in performance can be observed the higher the number of instances dimensions and clusters. Due to the poor results of Visual C++ we omit further measurements and concentrate on the results produced by the Intel C++ version.

For the fully optimized Intel C++ version the speedups are obviously smaller as this version makes use of the SIMD instruction-set of the CPU. A speedup by a factor of 1.5 to 14 can be observed for all but the smallest data set on System 1. Interestingly this version performs better for lower dimensionality for high instance counts. This is due to the fact that as the centroid update time decreases due to optimization the transfer time starts to play a bigger role. Nevertheless there is still a considerable speedup observable.

The diagrams in figure 7 also explain why the GPU-based implementation does not match the CPU implementation for very small data sets. From the plot it can be seen that for 500 data points nearly all the time is spent on the GPU. This time span also includes the calling overhead for invoking the GPU labeling stage. This invocation time actually takes longer than labeling the data items.

The GPU-based implementation is clearly memory bound as there are more memory accesses than floating point operations. Figure 8 shows the throughput achieved by the GPU for various dataset sizes. A peak throughput of 44GB/s could be achieved for the largest problem with 500000 instances

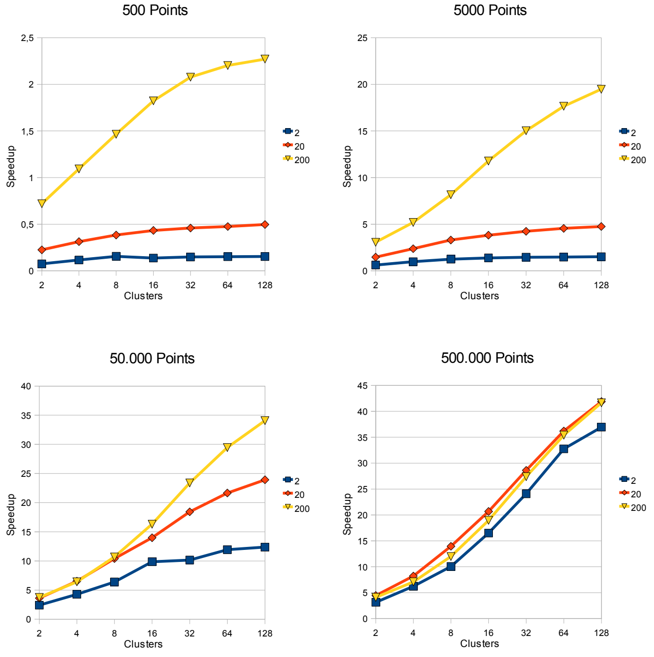


Figure 5. Speedup measured against the Visual C++ compiler for various instance counts and dimensions on System 1

with 200 dimensions and 128 clusters. For the used hardware the peak performance is given as 57.6 GB/s. Therefore we are highly confident that the implementation is nearly optimal. For very small datasets the throughput is not even one percent of the reachable peak performance. Interestingly, for the dataset with 500 instances, 200 dimensions and 128 clusters a sharp deviation from an otherwise log like curve can be observed. This indicates a sweetspot at which the graphics card is able to benefit from the data size.

Due to being memory bound the GFLOP counts do of course not reach the hardwares peak values. Figure 9 shows the approximate GFLOP/s achieved by our implementation. As with memory transfers the performance gets better the more data is thrown at the hardware. Also, the spike in the small dataset that was found in the memory throughput analysis can also be found here for the same reason as stated above: given a certain dataset size the graphics card is better able to benefit from its inherent parallel nature.

We repeated the measurements for System 2. Figures 10 through 13 present the results from System 2. The relative performance compared to the CPU version does not increase immensely. However, one has to take into account the the CPU used on System 2 is also better than the one used for relative measurements on System 1. The trend is very similar to the one observed on System 1. We achieve near peak memory throughput but can not exploit the computational capabilities fully. Additionally the performance decreases a

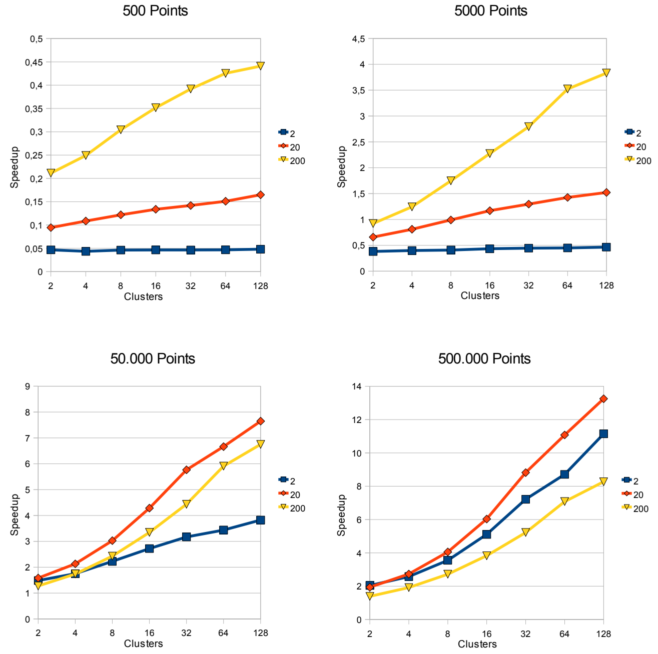


Figure 6. Speedup measured against Intel C++ compiler for various instance counts and dimensions on System 1

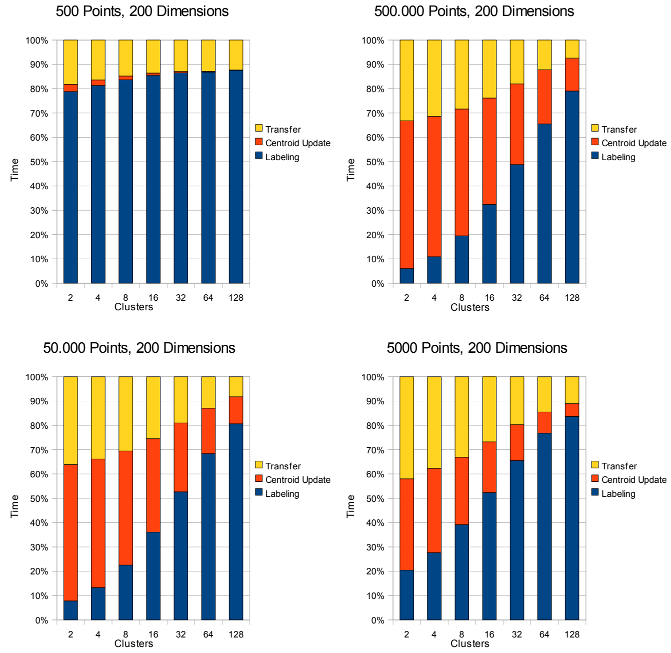


Figure 7. Percentage of time used for the different stages on the GPU on System 1

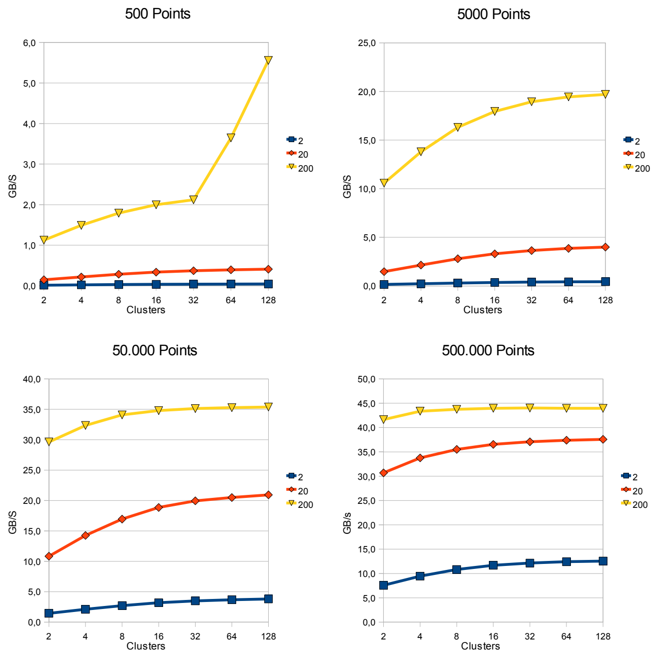


Figure 8. Memory throughput for various instance counts and dimensions in Gigabytes on System 1

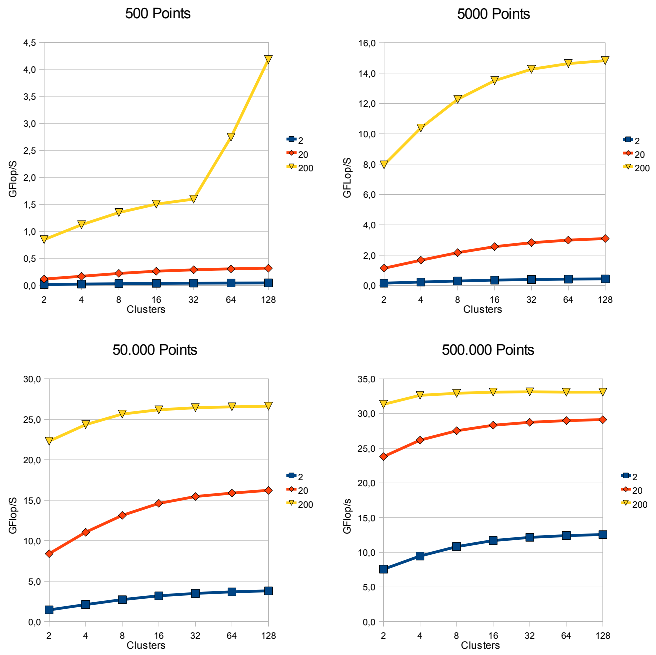


Figure 9. GFLOP/s reached for various instance counts and dimensions on System 1

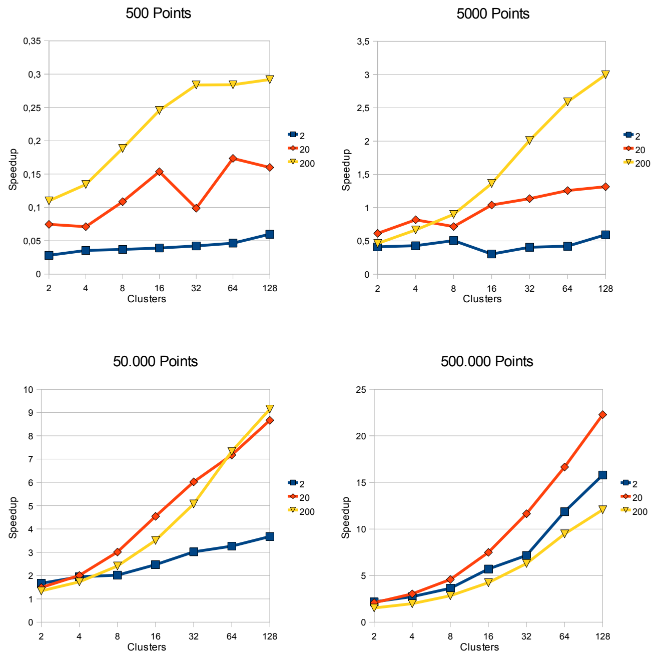


Figure 10. Speedup measured against Intel C++ compiler for various instance counts and dimensions on System 2

little as the number of clusters and dimensions increases for the largest dataset. We do not have an explanation for this behaviour and will investigate this in future work. However, from the timing chart it is clear the the GPU is doing a lot less work in total compared to the work it did in System 1. Transfertimes play a much bigger role overall and the centroid update stage also takes up a bigger amount of time. It seems therefore advisable to keep the GPU saturated with more data in order to increase its overall contribution to performance.

For some test runs slight variations in the resulting centroids were observed. These variations are due to the use of combined multiplication and addition operations (MADD) that introduce rounding errors. Quantifying these errors was out of the scope of this work, especially as no information from the vendor on the matter was available.

VII. Conclusion & Future Work

Exploiting the GPU for the labeling stage of k-means proved to be beneficial especially for large data sets and high cluster counts. The presented implementation is only limited in the available memory on the GPU and therefore scales well. However, some drawbacks are still present. Many real-life data sets like document collections operate in very high dimensional spaces where document vectors are sparse. The implementation of linear algebra operations on sparse data on the GPU has yet to be solved optimally. Necessary access patterns such as memory coalescing make

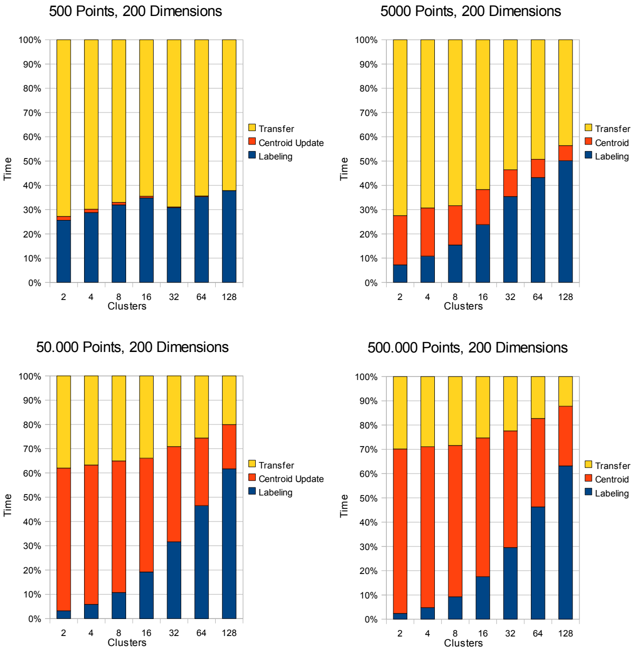


Figure 11. Percentage of time used for the different stages on the GPU on System 1

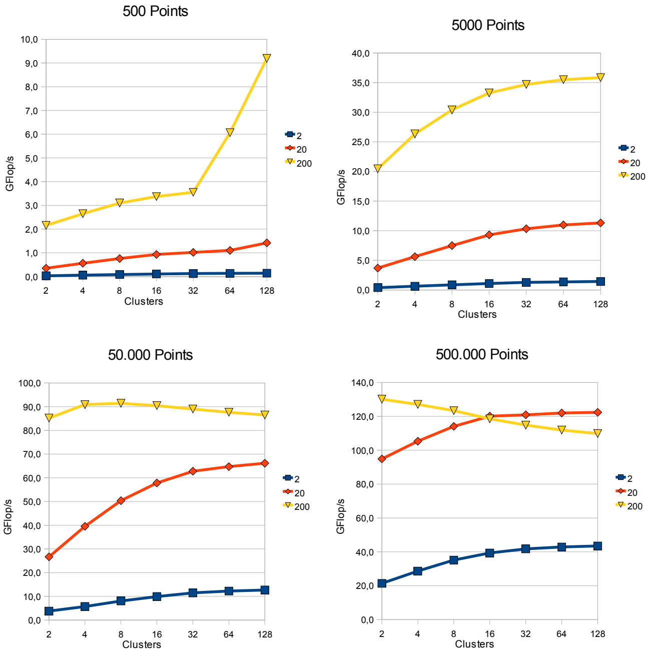


Figure 12. Memory throughput for various instance counts and dimensions in Gigabytes on System 1

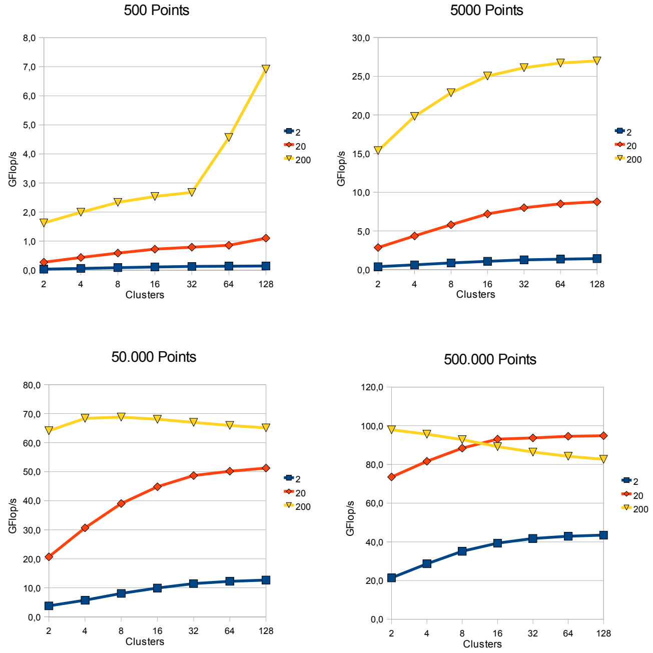


Figure 13. GFLOP/s reached for various instance counts and dimensions on System 1

this a very hard undertaking. Also, the implementation presented is memory bound meaning that not all of the GPUs computational power is harvested. Finally, due to rounding errors the results might not equal the results obtained by a pure CPU implementation. However, our experimental experience showed that the error is negligible.

Future work will involve experimenting with other k-means variations such as spherical or kernel k-means that promise to increase the computational load and therefore better suit the GPU paradigm. Also, an efficient implementation of the centroid update stage on the GPU will be investigated.

REFERENCES

[1] Mario Zechner and Michael Granitzer. Accelerating k-means on the graphics processor via cuda. *Intensive Applications and Services, International Conference on*, 0:7–15, 2009.

[2] A. K. Jain, M. N. Murty, and P. J. Flynn. Data clustering: a review. *ACM Comput. Surv.*, 31(3):264–323, September 1999.

[3] Jian Yi, Yuxin Peng, and Jianguo Xiao. Color-based clustering for text detection and extraction in image. In *MULTIMEDIA '07: Proceedings of the 15th international conference on Multimedia*, pages 847–850, New York, NY, USA, 2007. ACM.

[4] Dannie Durand and David Sankoff. Tests for gene clustering. In *RECOMB '02: Proceedings of the sixth*

annual international conference on Computational biology, pages 144–154, New York, NY, USA, 2002. ACM.

[5] Adil M. Bagirov and Karim Mardaneh. Modified global k-means algorithm for clustering in gene expression data sets. In *WISB '06: Proceedings of the 2006 workshop on Intelligent systems for bioinformatics*, pages 23–28, Darlinghurst, Australia, Australia, 2006. Australian Computer Society, Inc.

[6] Shi Zhong. Efficient streaming text clustering. *Neural Netw.*, 18(5-6):790–798, 2005.

[7] Stuart P. Lloyd. Least squares quantization in pcm. *IEEE Transactions on Information Theory*, 28(2):129–136, 1982.

[8] J. B. MacQueen. Some methods for classification and analysis of multivariate observations. In L. M. Le Cam and J. Neyman, editors, *Proc. of the fifth Berkeley Symposium on Mathematical Statistics and Probability*, volume 1, pages 281–297. University of California Press, 1967.

[9] David Arthur and Sergei Vassilvitskii. k-means++: the advantages of careful seeding. In Nikhil Bansal, Kirk Pruhs, and Clifford Stein, editors, *SODA*, pages 1027–1035. SIAM, 2007.

[10] Jens Krüger and Rüdiger Westermann. Linear algebra operators for gpu implementation of numerical algorithms. In *SIGGRAPH '03: ACM SIGGRAPH 2003 Papers*, pages 908–916, New York, NY, USA, 2003. ACM.

[11] Ian Buck, Tim Foley, Daniel Horn, Jeremy Sugerman, Kayvon Fatahalian, Mike Houston, and Pat Hanrahan. Brook for gpus: stream computing on graphics hardware. In *SIGGRAPH '04: ACM SIGGRAPH 2004 Papers*, pages 777–786, New York, NY, USA, 2004. ACM.

[12] Mark Harris. Mapping computational concepts to gpus. In *SIGGRAPH '05: ACM SIGGRAPH 2005 Courses*, page 50, New York, NY, USA, 2005. ACM.

[13] Nvidia cuda site, 2007. http://www.nvidia.com/object/cuda_home.html.

[14] Ati close to metal guide, 2007. http://ati.amd.com/companyinfo/researcher/documents/ATI_CTM_Guide.pdf.

[15] NVIDIA. *NVIDIA CUDA Programming Guide 2.0*. 2008. http://developer.download.nvidia.com/compute/cuda/2_0/docs/NVIDIA_CUDA_Programming_Guide_2.0.pdf.

[16] P. Drineas, A. Frieze, R. Kannan, S. Vempala, and V. Vinay. Clustering large graphs via the singular value decomposition. *Mach. Learn.*, 56(1-3):9–33.

[17] Leon Bottou and Yoshua Bengio. Convergence properties of the K-means algorithms. In G. Tesauro, D. Touretzky, and T. Leen, editors, *Advances in Neural Information Processing Systems*, volume 7, pages 585–592. The MIT Press, 1995.

[18] Inderjit S. Dhillon and Dharmendra S. Modha. A data-clustering algorithm on distributed memory multiprocessors. In *Large-Scale Parallel Data Mining, Lecture Notes in Artificial Intelligence*, pages 245–260, 2000.

[19] Hiroyuki Takizawa and Hiroaki Kobayashi. Hierarchical parallel processing of large scale data clustering on a pc cluster with gpu co-processing. *J. Supercomput.*, 36(3):219–234, 2006.

[20] Jesse D. Hall and John C. Hart. Gpu acceleration of iterative clustering. Manuscript accompanying poster at GP2: The ACM Workshop on General Purpose Computing on Graphics Processors, and SIGGRAPH 2004 poster (2004).

[21] Feng Cao, Anthony K. H. Tung, and Aoying Zhou. Scalable clustering using graphics processors. In *WAIM*, pages 372–384, 2006.

[22] David Luebke and Greg Humphreys. How gpus work. *Computer*, 40(2):96–100, 2007.

[23] John Nickolls, Ian Buck, Michael Garland, and Kevin Skadron. Scalable parallel programming with cuda. In *SIGGRAPH '08: ACM SIGGRAPH 2008 classes*, pages 1–14, New York, NY, USA, 2008. ACM.

A Virtualized Infrastructure for Automated BitTorrent Performance Testing and Evaluation

Răzvan Deaconescu George Milescu Bogdan Aurelian Răzvan Rughiniş
Nicolae Ţăpuş
University Politehnica of Bucharest
Computer Science Department
Splaiul Independenţei nr. 313, Bucharest, Romania
{*razvan.deaconescu, george.milescu, bogdan.aurelian, razvan.rughinis, nicolae.tapus*}*@cs.pub.ro*

Abstract

In the last decade, file sharing systems have generally been dominated by P2P solutions. Whereas email and HTTP have been the "killer apps" of the earlier Internet, a large percentage of the current Internet backbone traffic is BitTorrent traffic [15]. BitTorrent has proven to be the perfect file sharing solution for a decentralized Internet, moving the burden from central servers to each individual station and maximizing network performance by enabling unused communication paths between clients.

Although there have been extensive studies regarding the performance of the BitTorrent protocol and the impact of network and human factors on the overall transfer quality, there has been little interest in evaluating, comparing and analyzing current real world implementations. With hundreds of BitTorrent clients, each applying different algorithms and performance optimization techniques, we consider evaluating and comparing various implementations an important issue.

In this paper, we present a BitTorrent performance evaluation infrastructure that we are using with two purposes: to test and compare current real world BitTorrent implementations and to simulate complex BitTorrent swarms. Our infrastructure consists of a virtualized environment simulating complete P2P nodes and a fully automated framework. For relevant use, different existing BitTorrent clients have been instrumented to output transfer status data and extensive logging information.

Keywords: BitTorrent; virtualization; automation; performance evaluation; client instrumentation

1 Introduction

P2P sharing systems are continuously developing and increasing in size. There is a large diversity of solutions and protocols for sharing data and knowledge which enable an increasing interest from common users and commercial and academic institutions [22].

It is assumed [15] that BitTorrent is responsible for a large portion of all Internet traffic. BitTorrent has proven to be the "killer" application of the recent ears, by dominating the P2P traffic in the Internet [24].

During the recent years BitTorrent [16] has become the de facto P2P protocol used throughout the Internet. A large portion of the Internet backbone is currently comprised of BitTorrent traffic [18]. The decentralized nature of the protocol insures scalability, fairness and rapid spread of knowledge and information.

As a decentralized system, a BitTorrent network is very dynamic and download performance is influenced by many factors: swarm size, number of peers, network topology, ratio enforcement. The innate design of the BitTorrent protocol implies that each client may get a higher download speed by unchoking a certain client. At the same time, firewalls and NAT have continuously been a problem for modern P2P systems and decrease the overall performance.

Despite implementing the BitTorrent specification [23] and possible extensions each client uses different algorithms and behaves differently on a given situation: it may limit the number of peers, it may use heuristic information for an optimistic unchoke, it could choose a better client to download from. An important point of consideration is the diversity and heterogeneity of peers in the Internet. Some peers have low bandwidth connections, some act behind NATs and firewalls, some use certain improvements to the protocol. These aspects make a thorough analysis of the protocol or of its implementations difficult as there is little to no control over the parameters in a real BitTorrent swarm.

The results presented in this paper are a continuation of previous work on BitTorrent applications as described at ICNS 2009 [1].

Our paper presents a BitTorrent performance evaluation infrastructure [30] that enables creating a contained environment for BitTorrent evaluation, testing various BitTorrent implementations and offers extensive status information about each peer. This information can be used for analysis, interpretation and correlation between different implementations and for analyzing the impact of a swarms state on the download performance.

In order to simulate an environment as real as possible, hundreds to thousands of computer systems are required, each running a particular BitTorrent implementation. Modern clusters could offer this environment, but the experiments require access to all systems, making the availability of such a cluster an issue.

The approach we propose in this paper is to use a virtualization solution to accommodate a close-to-real-world testing environment for BitTorrent applications at a fraction of the costs of a real hardware solution (considering the number of computer systems). Our virtualization solution uses the lightweight OpenVZ [21] application that enables fast creation, limited execution overhead and low resource consumption. In this paper we show that, by using commodity hardware and OpenVZ, a virtual testing environment can be created with at least ten times more simulated systems than the real one used for deployment.

On top of the virtualized infrastructure, we developed a fully automated BitTorrent performance evaluation framework. All tested clients have been instrumented to use command line interfaces that enable automated actions. Clients are started simultaneously and results are collected after the simulation is complete.

The paper is organized as follows: Section 2 provides background information, keywords and acronyms used throughout the article, Section 3, 4, 5 present the infrastructure and framework used for our BitTorrent experiments; Section 6 and 7 describe OpenVZ and MonALISA, the virtualization and monitoring solution we used; we present the experimental setups and results of various experiments in Section 8; Section 9 describes the web interface architecture built on top of the framework; Section 10 and 11 present concluding remarks and related work.

2 Background

Our paper deals with recent concepts related to peer-to-peer networks, BitTorrent in particular, and virtualization. This section gives some definitions of terms used throughout the paper.

P2P networks are part of the peer-to-peer paradigm. Each peer is simultaneously a client and a server. P2P networks are decentralized systems sharing information and bandwidth as opposed to the classical centralized client-server paradigm.

BitTorrent is the most used P2P protocol in the Internet. Since its creation by Bram Cohen in 2001, BitTorrent has proven to provide the best way to allow file distribution among its peers. The BitTorrent protocol makes a separation between a file's content and its metadata. The metadata is stored in a specialized **.torrent file**. The .torrent file stores piece information and hashes and tracker information (see below) and is usually distributed through the use of a web server. BitTorrent is not a completely decentralized protocol. A special server, called **tracker** is used to intermediate initial connections between peers.

A set of peers sharing a particular file (i.e. having access and using the same .torrent file) are said to be part of the same **swarm**. A tracker can mediate communication in multiple swarms at the same time. Each peer within a swarm is either a seeder of a leecher. A **seeder** is a peer who has complete access to the shared file; the seeder is only uploading. For a swarm to exist there has to be an initial seeder with access to the complete file and its associated metadata in the .torrent file. A peer is a **leecher** as long as it has only partial access to the file (i.e. it is still downloading). A healthy swarm must contain a good number of seeders.

There is a great variety of BitTorrent clients, some of which have been the subject of the experiments described in this paper. There are also BitTorrent libraries (such as libtorrent-rasterbar or libtorrent-rakshasa) that form the basis for particular BitTorrent implementations. Some of the more popular clients are uTorrent, Azureus, Transmission, rTorrent, BitTorrent (the official BitTorrent client, also known as Mainline).

Our paper describes the use of virtualization technology in the benefit of simulating partial or complete BitTorrent swarms. For our experiments we have used the **OpenVZ** [21] virtualization solution. OpenVZ is an operating-system level virtualization solution. This means that each virtual machine (also known as **VE** - *virtual environment*) that it will run on the same kernel as the host system. This approach has the advantage of using a small amount of resources for virtual machine implementation. Each OpenVZ VE uses a part of the

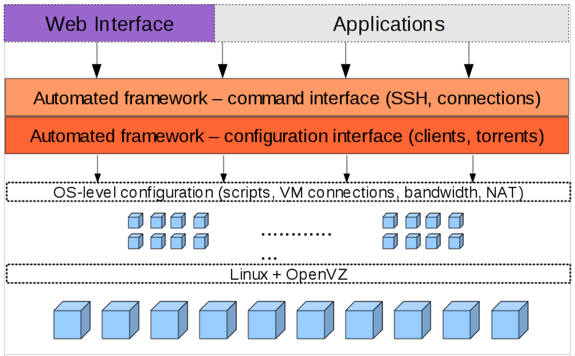

Web Interface
Applications
Automated framework – command interface (SSH, connections)
Automated framework – configuration interface (clients, torrents)
OS-level configuration (scripts, VM connections, bandwidth, NAT)
Linux + OpenVZ

Figure 1. Overall Architecture of BitTorrent Performance Evaluation Infrastructure

host file-system and each OpenVZ process is a process in the host system.

Each host system acts as a gateway for the OpenVZ VEs. For our infrastructure, the host system uses **NAT** (*Network Address Translation*) to allow communication between and OpenVZ VE and the outside world. The current setup uses **SNAT** (*Source NAT*) to enable access from the OpenVZ VEs to the outside world and **DNAT** (*Destination NAT*) to enable connections **to** the virtual machines, for uploading.

As the base system is a Linux-based distribution NAT is handled through the use of `iptables`, the tool that handles packet filtering and manipulation. Our framework also uses **tc** (*traffic control*) for traffic limitation.

SSH (*Secure Shell*) is the basic communication protocol for commanding VEs and BitTorrent clients. It has the advantage of being secure and allowing easy communication with VEs. At the same time, it allows automating commands through a scripted interface. As the framework uses **CLI** (*Command Line Interface*) for automating commands, the use of SSH is very helpful.

Our measurements use MonALISA [20] for real time monitoring of BitTorrent parameters, usually download speed and download percentage. MonALISA provides a specialized infrastructure for storing and visualizing required parameters. A lightweight monitoring agent is used on each OpenVZ VE to send periodic updates to the MonALISA repository.

3 Overall architecture

Figure 1 presents a general overview of the BitTorrent performance evaluation infrastructure.

The infrastructure consists of commodity hardware systems running GNU/Linux. Each system uses OpenVZ virtual machine implementation to run multiple virtual systems on the same hardware node.

Each virtual machine contains the basic tools for running and compiling BitTorrent clients. Tested BitTorrent implementations have been instrumented for automated command and also for outputting status and logging information required for subsequent analysis and result interpretation. As the infrastructure aims to be generic among different client implementations, the addition of a new BitTorrent client resumes only at adding the required scripts and instrumentation.

Communication with the virtual machines is enabled through the use of DNAT and iptables. TC (traffic control) is used for controlling the virtual links between virtual systems.

Each virtual machine uses a set of scripts to enable starting, configuration, stopping and result gathering for BitTorrent clients.

A test/command system can be used to start a series of clients automatically through the use of a scripted interface. The command system uses SSH to connect to the virtual machines (SSH is installed and communication is enabled through DNAT/iptables) and command the BitTorrent implementations. The SSH connection uses the virtual machine local scripts to configure and start the clients.

The user can directly interact with the automated framework through the use of command scripts, can embed the commanding interface in an application or can use the web interface. The web interface was developed to facilitate the interaction between the user, the BitTorrent clients and the virtual machines. It enables most of the actions that an user is able to accomplish trough direct use of the command scripts.

4 BitTorrent Clients

For our experiments we selected the BitTorrent clients that are most significant nowadays, based on the number of users, reported performance, features and history.

We used Azureus, Tribler, Transmission, Aria, libtorrent rasterbar/hrktorrent, BitTornado and the mainline client (open source version). All clients are open source as we had to instrument them to use a command line interface and to output verbose logging information.

Azureus, now called Vuze, is a popular BitTorrent client written in Java. We used Azureus version 2.3.0.6. The main issue with Azureus was the lack of a proper CLI that would enable automation. Though limited, a “Console UI” module enabled automating the tasks of running Azureus and gathering download status and logging information.

Tribler is a BitTorrent client written in Python and one of the most successful academic research projects. Developed by a team in TU Delft, Tribler aims at adding various features to BitTorrent, increasing download speed and user experience. We used Tribler 4.2. Although a GUI oriented client, Tribler offers a command line interface for automation. Extensive logging information is enabled by updating the value of a few variables.

Transmission is the default BitTorrent client in the popular Ubuntu Linux distribution. Transmission is written in C and aims at delivering a good amount of features while still keeping a small memory footprint. The version we used for our tests was transmission 1.22. Transmission has a fully featured CLI and was one of the clients that were very easy to automate. Detailed debugging information regarding connections and chunk transfers can be enabled by setting the TR_DEBUG_FD environment variable.

Aria2 is a multiprotocol (HTTP, FTP, BitTorrent, Metalink) download client. Throughout our tests we used version 0.14. aria2 natively provides a CLI and it was easy to automate. Logging is also enabled through CLI arguments. Aria2 is written in C++.

libtorrent rasterbar/hrktorrent is a BitTorrent library written in C++. It is used by a number of BitTorrent clients such as Deluge, BitTorrent and SharkTorrent. As we were looking for a client with a CLI we found hrktorrent to be the best choice. hrktorrent is a lightweight implementation over rasterbar libtorrent and provides the necessary interface for automating a BitTorrent transfer, although some modifications were necessary. Rasterbar libtorrent provides extensive logging information by defining the TORRENT_LOGGING and TORRENT_VERBOSE_LOGGING_MACROS. We used version 0.13.1 of rasterbar libtorrent and the most recent version of hrktorrent.

BitTornado is an old BitTorrent client written in Python. The reason for choosing it to be tested was because of a common background with Tribler. However, as testing revealed, it had its share of bugs and problems and it was eventually dropped.

BitTorrent Mainline is the original BitTorrent client written by Bram Cohen in Python. We used version 5.2 during our experiments, the last open-source version. The mainline client provides a CLI and logging can be enabled through minor modifications of the source code.

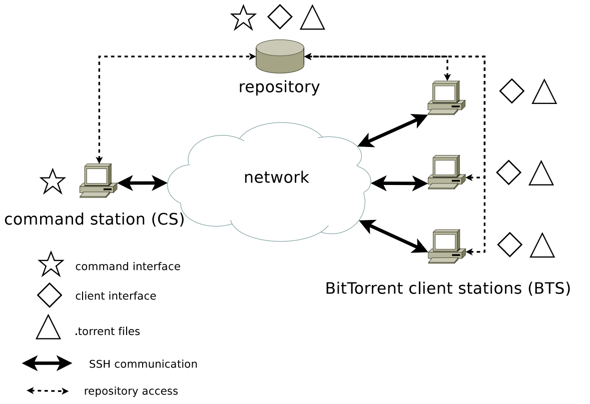


Figure 2. Client Control Infrastructure

5 Framework

As shown in Figure 2, the client control infrastructure on top of which our framework runs consists of a command station, a repository and a set of client stations. Through the use of OpenVZ [21], the client stations are virtual machines simulating common features of a standard computer system.

The current infrastructure is located in NCIT Cluster [29] in University Politehnica of Bucharest. We are using commodity hardware systems as described in Section 8. The systems we are using for BitTorrent/P2P experiments are located in the same local area network. Each system uses a 1 Gbit Ethernet link and the cluster provides an 10 Gbit external link. Each system is able to sustain a 25 MB/s download speed from external download sources (i.e. not in cluster).

Communication between the OpenVZ virtual machines is handled by the host operating system, which acts as a gateway enabling communication with other virtual machines.

Communication between the CS (Command Station) and the BTS (BitTorrent Client Station) is handled over SSH for easy access and commanding. The repository is used to store the control framework implementation and .torrent files for download sessions. Each BTS checks out from the repository the most recent framework version and the .torrent files to be used.

The control framework is actually a set of small shell scripts enabling the communication between the CS and the BTS and running each BitTorrent client through its command line interface.

For each download session the commander specifies the .torrent file to be used and the mapping between each BitTorrent client and a BTS. Through SSH a specific script is being run on the target BTS enabling the BitTorrent client. The BTS doesn't need to be in the same network. The only requirement for the client stations is to run an SSH server and be accessible through SSH.

Each BitTorrent client uses a specialized environment for storing logging and download status information. All this data can then be collected and analyzed.

5.1 Repository access

6 Using a virtualized environment

Various extensions, sharing ratio enforcement policies and moderation techniques have been deployed to improve the overall efficiency of the BitTorrent protocol. An important point to consider is the diversity and heterogeneity of peers in the Internet. Some peers have low bandwidth connections, some act behind NATs and firewalls, some use certain improvements to the protocol. These aspects make a thorough analysis of the protocol or of its implementations difficult as there is little to no control over the parameters in a real BitTorrent swarm.

A solution is creating a contained environment for BitTorrent evaluation. However, in order to simulate an environment as real as possible, hundreds to thousands of computer systems are required, each running a particular BitTorrent implementation. Modern clusters could offer this environment, the experiments require access to all required systems, making the availability of such a cluster an issue.

The approach we propose is to use a virtualization solution to accommodate a close-to-real-world testing environment for BitTorrent applications at a fraction of the costs of a real hardware solution (considering the number of computer systems). Our virtualization solution uses the lightweight OpenVZ [21] application that enables fast creation, limited execution overhead and low resource consumption. In this paper we show that, by using commodity hardware and OpenVZ, a virtual testing environment can be created with at least ten times more simulated systems than the real one used for deployment.

Creating a virtualized environment requires the hardware nodes where virtual machines will be deployed, the network infrastructure, a set of OpenVZ templates for installation and a framework that enables commanding clients inside the virtual machines.

Each virtual machine runs a single BitTorrent application that has been instrumented to use an easily-automated CLI.

6.1 OpenVZ

OpenVZ [21] is an operating system-level virtualization solution. It can run only a Linux virtual environment over an OpenVZ-enabled Linux kernel. A virtual machine is also called a container or virtual environment (VE). OpenVZ is a lightweight virtualization solution incurring minimal overhead compared to a real environment.

OpenVZs advantages are low-resource consumption and fast creation times. As it is using the same kernel as the host system, OpenVZs memory and CPU consumption is limited. At the same time, OpenVZ file-system is a sub-folder in the hosts file-system enabling easy deployment and access to the VE. Each VE is thus a part of the main file-system and can be encapsulated in a template for rapid deployment. One simply has to uncompress an archive, edit a configuration file and setup the virtual machine (host name, passwords, network settings).

OpenVZs main limitation is the environment in which it runs: the host and guest systems must both be Linux. At the same time certain kernel features that are common in a hardware-based Linux system are missing: NFS support, NAT, etc., due to inherent design.

Despite its limitations, OpenVZ is the best choice for creating a virtualized environment for the evaluation of BitTorrent clients. Its minimal overhead and low-resource consumption enables running tens of virtual machines on the same hardware node with little penalty.

7 Monitoring with MonALISA

Deploying a large number of virtual environments implies gathering important amount of information for analysis and interpretation. While status and logging information is gathered and stored for each experimental session, we enabled the use of the MonALISA [20] client for real time monitoring and data storage.

MonALISA uses a distributed infrastructure to monitor various experiments and activities. It has a diversity of features ranging from easily integrated API, real time monitoring, graphical representation, data storage for further use, etc.

We extended our framework to use MonALISA for real time monitoring of download speed and other factors. Each client can be configured to send data to a MonALISA agent. The MonALISA agent parses that data and creates a close-to-real-time graphical evolution of the download speed. There is a small delay

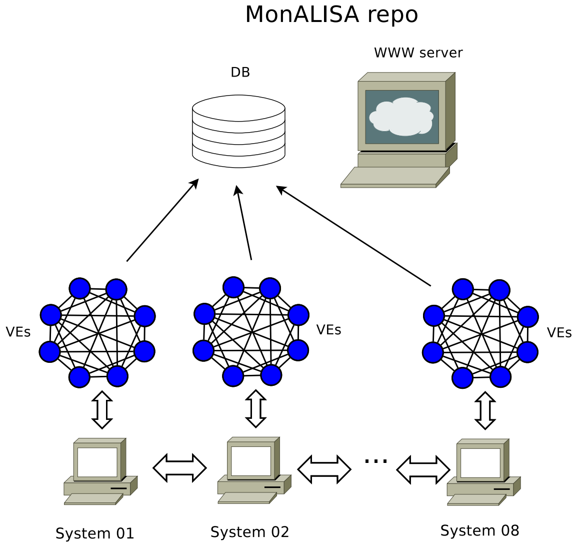


Figure 3. MonALISA Monitoring Infrastructure

between the client collecting sufficient data and sending it and the MonALISA agent processing it.

Figure 3 is an overview of the monitoring infrastructure involving MonALISA. All virtualized systems are able to send data to a MonALISA server and subsequently to a MonALISA repository. A web interface application uses data from the repository to publish it as history or real time graphs. Besides the web interface, MonALISA offers an interactive client that must be installed on the user system. The client enables read access to information in different clusters and can be used to create on-demand graphical representation of measured features.

8 Experimental setups

We used two major experimental setups. The first setup has been used before the addition of the virtualized environment and was dedicated to measurements and comparisons between different BitTorrent clients. The second setup is the current one and uses two benefits of the virtualized environment to simulate complete swarms: client station characteristics and network interconnections.

8.1 Performance evaluation experiment setup

Our first experimental setup consisted of six identical computers running the same operating system and software. The hardware configuration includes Pentium 4 2GHz CPU, 1 GB RAM, 160 GB HDD. We used Ubuntu 7.10 Gutsy Gibbon as an operating system.

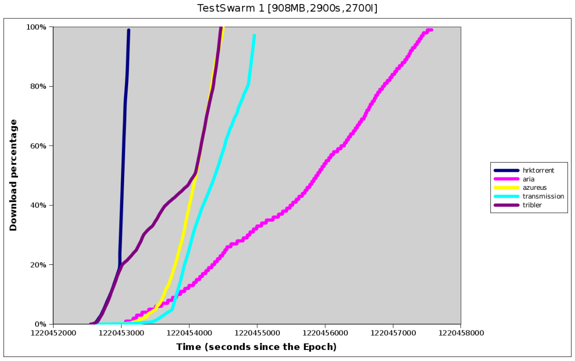


Figure 4. Test Swarm 1

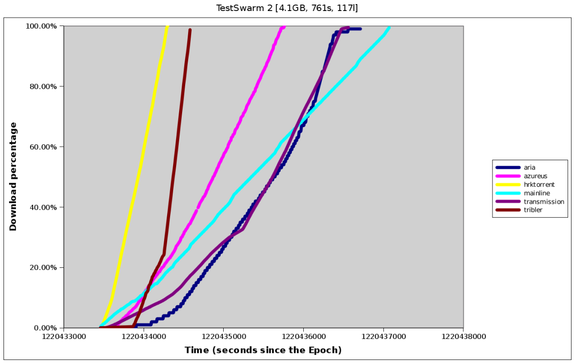


Figure 5. Test Swarm 2

The six computers are connected in the same network, thus using common bandwidth to access the Internet. The computers have been firewalled from each other such that communication is enabled only with peers from the Internet.

Most of our experiments were simultaneous download sessions. Each system ran a specific client in the same time and conditions as the other clients. Results and logging data were collected after each client finished its download.

8.1.1 Results

Our framework was used to test different swarms (different .torrent files). Most scenarios involved simultaneous downloads for all clients. At the end of each session, download status information and extensive logging and debugging information were gathered from each client. Figure 4 and figure 5 are comparisons between different BitTorrent clients running on the same environment in similar download scenarios. The graphical representation show the download ratio/percentage evolution with respect to time.

The first test runs simultaneous sessions for five clients (hrktorrent, aria, azureus, transmission, tribler) part of the same swarm. The swarm uses 2900 seeders and 2700 leechers and a 908MB file. All clients were started at the same time on different systems in the same network. The operating system, hardware and network characteristics are identical for all clients. Aria is the clear loser of the race. A tight competition is going between hrktorrent and Tribler at the beginning of the download session. Both clients display a high acceleration (increase in download speed). However, Tribler's speed/download rate is gets lower at around 20% of the download and hrktorrent comes out as the clear winner. Tribler and Azureus finish their download at around the same time, with Transmission coming out fourth, and Aria last.

The second test also runs simultaneous sessions for all clients presented in Section 4. The current swarm uses a 4.1 GB file, 761 seeders and 117 leechers. hrktorrent again displays an excellent start, with all the other clients lagging. Tribler starts a bit late, but manages to catch up and, at about 25% of the download size, is faster than hrktorrent. At the end of the session, the first three clients are the same (hrktorrent, Tribler, Azureus) with Transmission, Aria and Mainline finishing last.

Table 1. Test Swarms Results

Client	Test1	Test2	Test3	Test 4
file size	908MB	4.1GB	1.09GB	1.09GB
seeders	2900	761	521	496
leechers	2700	117	49	51
aria2c	1h17m	53m53s	8m	10m23s
azureus	32m41s	38m33s	N/A	7m
bittorrent	4h53m	60m39s	26m	14m
libtorrent	**9m41s**	**15m13s**	**2m30s**	**2m14s**
transmission	40m46s	53m	7m	5m
tribler	34m	21m	N/A	N/A

Table 1 presents a comparison of the BitTorrent clients in four different scenarios. Each scenario means a different swarm. Although much data was collected, only the total download time is presented in the table.

Due to bugs with the Tribler and Azureus clients, some results are missing and are marked with **N/A** in Table 1. The two clients did continue their download but at a negligible speed and they were stopped.

The conclusions drawn after result analysis were:

- hrk/libtorrent is continuously surpassing the other clients in different scenarios;
- tribler, azureus and transmission are quite good clients but lag behind hrktorrent;
- mainline and aria are very slow clients and should be dropped from further tests;
- swarms that are using sharing ratio enforcement offer better performance; a file is downloaded at least 4 times faster within a swarm using sharing ratio enforcement.

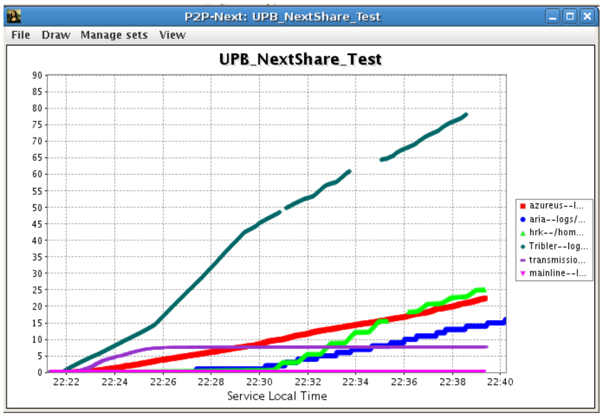


Figure 6. Close-to-real-time Monitoring

8.1.2 Integration with MonALISA

Figure 6 is a screenshot of a download session rendered using MonALISA. By using a MonALISA client, a close-to-real-time evolution of a BitTorrent download session can be obtained.

8.2 Virtualized experimental setup

Our current setup consists of 8 computers each running 5 OpenVZ virtual environments (VEs). All systems are identical with respect to the CPU power, memory capacity and HDD space and are part of the same network. The network connections are 1Gbit Ethernet links.

The hardware configuration for each system includes:

- 2GB RAM
- Intel(R) Pentium(R) 4 CPU 3.00GHz dual-core
- 300GB HDD

All systems are running the same operating system (Debian GNU/Linux Lenny/testing) and the same software configuration.

8.3 Resource consumption

With all 5 VEs active and running a BitTorrent client in each of them, memory consumption is 180MB-250MB per system. With no VE running, the memory consumption is around 80MB-110MB. The amount of memory used by the VEs is:
$VEmem = sVEmem - smem$
where:

- $VEmem$ is the memory consumption of VEs
- $sVEmem$ is the memory consumption of base-system and VEs
- $smem$ is the memory consumption by the "bare-bones" base-system

This means that the minimum and maximum values of memory consumption by the VEs are:
$min\_VEmem = min\_sVEmem - max\_smem$
$max\_VEmem = max\_sVEmem - min\_smem$

Given the above values, it results that the 5 VEs use between 70MB and 170MB of RAM or a rough estimate of 15MB to 35MB per VE.

The BitTorrent client in use is hrktorrent, a libtorrent-rasterbar based implementation. hrktorrent is a light addition to the libtorrent-rasterbar library, so memory consumption is quite small while still delivering good performance. On average, each virtual machine uses at most 40MB of RAM when running hrktorrent in a BitTorrent swarm.

The current partitioning scheme for each system leaves 220GB of HDD space for the VEs. However, one could upgrade that limit safely to around 280GB. Each basic complete VE (with all clients installed) uses 1.7GB of HDD space. During a 700MB download session, each client outputs log files using 30-50MB of space.

Processor usage is not an issue as BitTorrent applications are mostly I/O intensive.

8.3.1 Scalability

As mentioned above, 5 active VEs running the hrktorrent client use about 70 to 250MB of RAM. This gives a rough estimate of about 15 MB to 35 MB of memory consumption per VE. As the basic system also uses at most 110MB of RAM, it results that the total memory consumption is at most $num\_VEs * 35MB + 110MB$.

From the HDD perspective, the basic system can be tuned to use 20GB of space with no major constraints on the software configuration. Each complete VE (able to run all CLI-based BitTorrent clients) uses 1.7GB of space. At the same time, 1GB of space should be left on each system for testing and logging purposes and 5GB for file transfer and storage. This means that about 8GB of space should be reserved for each VE.

The above values are a rough estimate. A carefully tuned system would manage to use less resources. However, we aimed to show that given these high values, an average PC could still sustain a significant amount of VEs with little overhead and resource penalty.

Table 2 gives an estimated maximum number of OpenVZ virtual environments a basic PC is able to run. **Bold font** means limitation is due to RAM capacity, while *italic font* means limitation is due to HDD space.

Table 2. Estimated Maximum Number of VEs per System

HDD Memory	1GB	2GB	4GB	8GB	16GB
80GB	*7*	*7*	*7*	*7*	*7*
120GB	*12*	*12*	*12*	*12*	*12*
200GB	*22*	*22*	*22*	*22*	*22*
300GB	**26**	*35*	*35*	*35*	*35*
500GB	**26**	**55**	*60*	*60*	*60*
750GB	**26**	**55**	*91*	*91*	*91*
1TB	**26**	**55**	**113**	*122*	*122*

The above mentioned values assume the usage of the hrktorrent/libtorrent BitTorrent client. They also assume the scheduling impact of all processes in the VEs induces low overhead on the overall performance. However, even considering the scheduling overhead, a modest system would still be able to run at least 10 to 20 VEs.

It can also be noticed that the primary limiting factor is the hard-disk, not the physical memory. However, given a large number of VEs the processing power also becomes important. Consequently the later numbers are realistic only with respect to memory and HDD, neglecting the context-switch overhead and CPU power.

We can safely conclude that a virtualized testing environment based on OpenVZ would provide similar testing capabilities as a non-virtualized cluster with at most 10% of the cost. Our experimental setup consisting of just 8 computers is able of running at least 100 virtualized environments with minimal loss of performance.

The virtualized environment is thus a cheaper and more flexible alternative to a full-fledged cluster, with little performance loss. Its main disadvantage is the asymmetry between virtualized environments that run on different hardware system. The main issue is network bandwidth between VEs running on the same hardware node and VEs running on different hardware nodes. This can be corrected by using traffic limitation ensuring a complete network bandwidth symmetry between the VEs.

8.3.2 Testing scenarios and results

Currently we are simulating swarms comprising of a single seeder and 39 initial leechers. 19 leechers are high bandwidth peers (512KB/s download speed, 256KB/s upload speed) and 20 leechers are low bandwidth peers (64KB/s download speed, 32KB/s upload speed).

The total time of an experiment involving all 40 peers and a 700MB CD image file is around 4 hours. It only takes about half an hour for the high bandwidth clients to download it.

We have been using Linux Traffic Control (TC) tool combined with iptables set-mark option to limit download and upload traffic to and from a VE.

Figure 7 and Figure 8 are real time representations of download speed evolution using MonALISA. The first figure shows the initial phase (first 10 minutes) of an experiment with the low bandwidth clients limited by the 64KB/s download speed line, and the high bandwidth clients running between 100KB/s and 450KB/s. The second figure presents the mid-phase of an experiment when high bandwidth clients finished downloading.

Figure 7 show the limitation of the low bandwidth peers while the high bandwidth peers have sparse download speed. Each high bandwidth client's speed usually follows an up-down evolution, and an increasing median as time goes by.

For the second swarm, at around 13:05, the high bandwidth clients have finished their download or are finishing in the following minutes, while the low bandwidth clients are still downloading. The high bandwidth clients have a large speed interval, while the low bandwidth clients are "gathered" around the 64KB limitation.

9 Web interface

In order to ease the use of the automated framework we developed a web interface that acts as a front-end to the scripted framework. Functionalities provided by the scripted framework are integrated within the web interface.

A web-based approach was chosen in favor of other possible interfaces because of certain advantages. An important advantage is accessibility: a browser is all

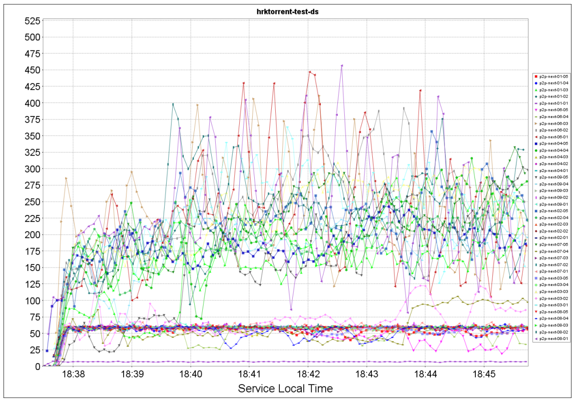


Figure 7. Download Speed Evolution (initial phase)

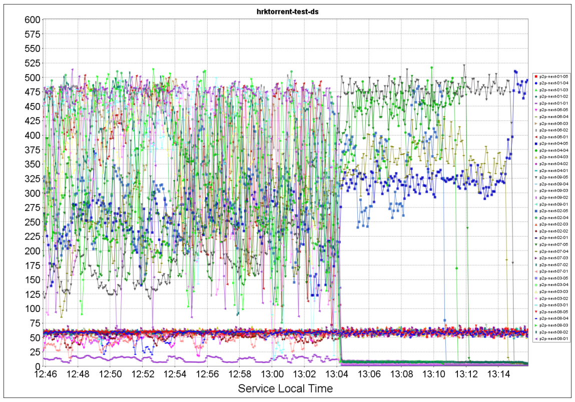


Figure 8. Download Speed Evolution (mid phase)

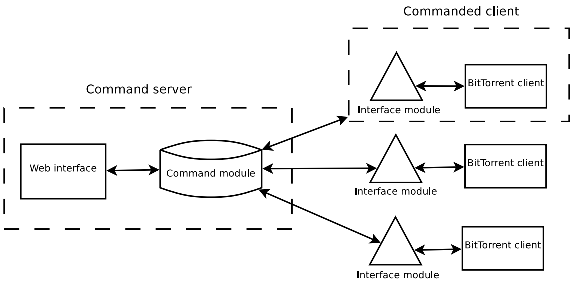


Figure 9. Web Interface Architecture

that is required in order to use the interface. This also means that the command server (see Figure 9) is installed on a single hardware system without further installation actions on client systems available to the user.

From a resource requirements perspective, the interface requires little (if any) complex graphical effects or additional software packages. The web technologies employed by the interface (HTTP, HTML, CSS, JavaScript) completely satisfy the requirements.

Another advantage is portability. The application, the web server and the tools employed are portable across different operating systems. From an interface point of view, this means that the application can be easily migrated from one system to another.

9.1 Web interface architecture

The web interface, as presented in Figure 9, is a front-end to the automated evaluation framework. More precisely, it is used as a substitute to the command station (CS) described in Figure 2. Command and control requests employed by the user in the CS can now be accessed through the use of the web interface. Communication between the web interface and the BitTorrent stations (BTS) is done, as in the case of the CS-BTS communication, through SSH. The web interface can be used to command, control, report client status and upload .torrent files to the client stations.

The web server used by the web interface is an Apache Tomcat server. The technologies employed are servlets, Java Server Pages, Java Struts and Java Tiles.

From the application perspective, the command server (web interface server in this case) sends commands to clients. The clients execute the commands/requests and return the error status.

The web interface enables the user to configure minimal administration of the client stations and the .torrent files. It can start and stop download sessions for BitTorrent clients, it can schedule starts and stops and it can report the current status for a specified client. For flexibility reasons the interface offers means through which the machines may be commanded individually or simultaneously, depending on the user's preference.

9.2 Integration with back-end framework

The commanding module consists of multiple subsystems, each fulfilling a specific task. It is responsible of keeping information about the client stations such as the availability information, connection details, available .torrent files, running client process, download sessions status and scheduled tasks.

Some data structures require periodic updates provided by the automated framework. The command module is also responsible of launching threads at specific times to complete these tasks.

The interface agent calls the BitTorrent client through the automated framework and sends it the parameters specified by the command module. Each specific action uses a different set of commanding parameters. Communication between the interface agent and the automated interface is handled through SSH.

10 Related work

While there are extensive studies and proposals regarding the internals of the BitTorrent protocol, there has been little concern about comparing and evaluating real world implementations. Guo et. al [6] developed an extensive study of P2P systems, but focusing on how the overall behavior of the swarm affects the performance. Pouwelse et. al [10] [11] have also gathered large amount of information and used it for detailing BitTorrent swarm behavior.

Most measurements and evaluations involving the BitTorrent protocol or BitTorrent applications are either concerned with the behavior of a real-world swarm or with the internal design of the protocol. There has been little focus on creating a self-sustained testing environment capable of using hundreds of controlled peers for gathering results and interpreting them.

Pouwelse et al. [11] have done extensive analysis on the BitTorrent protocol using large real-world swarms, focusing on the overall performance and user satisfaction. Guo et. al [5] have modeled the BitTorrent protocol and provided a formal approach to the its functionality. Bharambe et. al [3] have done extensive studies on improving BitTorrents network performance.

Iosup et al. [7] used a testing environment involving four major BitTorrent trackers for measuring topology and path characteristics. They used nodes in Planet-

Lab. The measurements were focused on geo-location and required access to a set of nodes in PlanetLab.

Garbacki et al. [4] have created a simulator for testing 2Fast, a collaborative download protocol. The simulator was useful only for small swarms that required control. Real-world experiments involved using real systems communicating with real BitTorrent clients in the swarm.

11 Conclusions and future work

The paper presents a BitTorrent performance evaluation infrastructure that uses virtualization to simulate hardware nodes and network interconnection features.

While there have been extensive studies regarding the performance of the BitTorrent protocol and how it can be improved by using carefully crafted algorithms, little attention has been given to analyzing and comparing real world implementations of the BitTorrent specifications. We created an easily deployable solution that enables automated testing of different BitTorrent clients in real world situations.

The proposed virtualized environment enables evaluation of the BitTorrent applications with only a fraction of the costs of a full-fledged cluster system. The advances of virtualization technologies and hardware systems ensure that complete experiments can be run in a virtualized environment with little penalty over a real environment.

Our approach makes use of the excellent OpenVZ virtualization software that incurs minimum overhead when creating and running virtualized environments. OpenVZ's low memory and HDD consumption enable tens of VEs, each running a BitTorrent client, to run on an average PC system (2GB RAM, 200GB HDD, 3GHz dual core CPU).

Given its flexibility, the virtualized environment can be used for a large variety of experiments and scales very well with respect to the number of simulated peers in a swarm. Our current experiments deal with low bandwidth/high bandwidth peers. We plan to extend these experiments to more peers, and to simulate a more dynamic swarm (with clients entering and leaving).

Our results identified the libtorrent-rasterbar implementation as the fastest client, clearly ahead of other implementations. We intend to analyze the logging output and investigate its source code to identify the clever tweaks that enable such an improvement over the other clients.

The current SSH communication means that BTS can be commanded only if its SSH server can be contacted. This makes it very difficult for clients that are behind NAT or firewalls to participate in a testing scenario. We aim to develop and deploy a server that accepts incoming connections regardless of NAT/firewall constraints and uses these connections to command BitTorrent clients.

Planned work is also to detect any potential preferences among clients, by enabling different BitTorrent implementations in a single closed swarm.

At the same time we intend to expand the reporting interface with information related to processor usage, memory consumption and number of connections for easy result interpretation and analysis. The MonALISA interface will also be extended for live reporting of various transfer parameters.

The virtualized environment gives easy access to modifying the number of seeders in a swarm, the number of firewalled clients, seeding time and many other variables. With full control over the entire swarm, one or more of the swarm parameters could be easily altered and then measure, analyze and interpret the results.

Our infrastructure uses real-world implementations of BitTorrent clients and a low-overhead virtualized testing environment. The virtualized testing environment is a novel approach that enables easy BitTorrent experiment creation, evaluation and analysis, and its flexibility allows potential extensions and improvements to be added resulting in better diversity over the experiments.

12 Acknowledgements

This paper is part of the research efforts within the P2P-Next FP7 project [22]. The code used within is project is open-source. Anyone can access its repository [30] or use the web interface [31] to browse the sources.

References

[1] Deaconescu, R., R. Rughiniş, N. Ţăpuş (2009). A BitTorrent Performance Evaluation Framework, In: Proceedings of ICNS 2009

[2] Deaconescu, R., R. Rughiniş, N. Ţăpuş (2009). A Virtualized Testing Environment for BitTorrent Applications, In: Proceedings of CSCS 17

[3] Bharambe, A. R., C. Herley, and V. N. Padmanabhan (2006). Analyzing and Improving a BitTorrent Network's Performance Mechanisms. In: Proceedings of Infocom'06

[4] Garbacki, P., A. Iosup, D. Epema, M. van Steen (2006). 2Fast: Collaborative Downloads in P2P Networks. In: Peer-to-Peer Computing, 23-30

[5] Guo, L., S. Chen, Z. Xiao, E. Tan, X. Ding, and X. Zhang (2005). Measurements, Analysis, and Modeling of BitTorrent-like Systems. In: Internet Measurement Conference

[6] Guo, L., S. Chen, Z. Xiao, E. Tan, X. Ding, and X. Zhang (2007). A Performance Study of BitTorrent-like Peer-to-Peer Systems. In: IEEE Journal on Selected Areas in Communications, Vol. 25, No. 1

[7] Iosup, A., P. Garbacki, J. Pouwelse, D. Epema (2006). Correlating Topology and Path Characteristics of Overlay Networks and the Internet. In: CCGRID

[8] Mol, J. J. D, J. A. Pouwelse, M. Meulpolder, D. H. J. Epema, and H. J. Sips (2007). Give-to-Get: An Algorithm for P2P Video-on-Demand

[9] Padala, P., X. Zhu, Z. Wang, S. Singhal, K. G. Shin (2007). Performance Evaluation of Virtualization Technologies for Server Consolidation. In: HPL-2007-59R1

[10] Pouwelse, J. A., P. Garbacki, D. H. J. Epema, and H. J. Sips (2004). A Measurement Study of the BitTorrent Peer-to-Peer File-Sharing System. In: Technical Report PDS-2004-003

[11] Pouwelse, J. A., P. Garbacki, D. H. J. Epema, and H. J. Sips (2005). The BitTorrent P2P file-sharing system: Measurements and Analysis. In: Fourth International Workshop on Peer-to-Peer Systems (IPTPS)

[12] Pouwelse, J. A., P. Garbacki, J. Wang, A. Bakker, J. Yang, A. Iosup, D. H. J. Epema, M. Reinders, M. R. van Steen, H. J. Sips (2005). Tribler: a social-based peer-to-peer system. In: Concurrency and Computation: Practice and Experience, Volume 20 Issue 2

[13] Vlavianos, A., M. Iliofotou, and M. Faloutsos (2006). BiToS: Enhancing BitTorrent for Supporting Streaming Applications

[14] Aria, The Fast and Reliable Download Utility (2009), http://aria2.sourceforge.net/, accessed 2009

[15] Arstechnica (2009), http://arstechnica.com/news.ars/post/20070903-p2p-responsible-for-as-much-as-90-percent-of-all-net-traffic.html, accessed 2008

[16] BitTorrent (2009), http://bittorrent.com, accessed 2009

[17] Hrktorrent (2009), http://50hz.ws/hrktorrent/, accessed 2009

[18] ipoque Internet studies, 2009, http://www.ipoque.com/resources/internet-studies/, accessed 2009

[19] Libtorrent (2009), http://www.rasterbar.com/products/libtorrent/, accessed 2009

[20] MonALISA, MONitoring Agents using a Large Integrated Services Architecture (2009), http://monalisa.cacr.caltech.edu, accessed 2009

[21] OpenVZ wiki (2009), http://wiki.openvz.org, accessed 2009

[22] P2P-Next (2009), http://www.p2p-next.org/, accessed 2009

[23] Theory Wiki (2009), http://wiki.theory.org/ BitTorrentSpecification, accessed 2008

[24] TorrentFreak (2009), http://torrentfreak.com/p2p-traffic-still-booming-071128/, accessed 2008

[25] TorrentFreak (2009), http://torrentfreak.com/bittorrent-launches-ad-supported-streaming-071218/, accessed 2008

[26] Transmission (2009), A Fast, Easy and Free BitTorrent Client (2009), http://www.transmissionbt.com/, accessed 2009

[27] Tribler (2009), http://www.tribler.org/trac, accessed 2009

[28] Vuze (2009), http://www.vuze.com/app, accessed 2009

[29] http://cluster.grid.pub.ro/, accessed 2009

[30] http://svn.tribler.org/abc/branches/razvan/perf

[31] http://www.tribler.org/browser/abc/branches/razvan/perf

The Influence of Group Members Arrangement on the Multicast Tree Cost

Maciej Piechowiak*, Maciej Stasiak†, Piotr Zwierzykowski†
**Kazimierz Wielki University, Bydgoszcz, Poland, e-mail: mpiech@ukw.edu.pl*
†*Poznań University of Technology, Poznań, Poland, e-mail: stasiak,pzwierz@et.put.poznan.pl*

***Abstract*—The article proposes a novel multicast routing algorithm without constraints and introduces the group members arrangement as a new parameter for analyzing multicast routing algorithms finding multicast trees. The objective of STA (Switched Trees Algorithm) is to minimize the total cost of the multicast tree using a modification of the classical Prim's algorithm (Pruned Prim's Heuristic) and the SPT (Shortest Path Tree) algorithm that constructs a shortest path tree between a source and each multicast node. In the article, the results of the proposed STA algorithm are compared with the representative algorithms without constrains. The results part of the article also contains some selected statistical properties of the multicast routing algorithms finding multicast trees as part of a wider research methodology.**

***Keywords*-multicasting, multicast tree, network topology, routing algorithm**

I. Introduction

The multicast technology is based on the simultaneous transmission of data concurrently to multiple destinations, from the source node to a group of destinations. Over the last few years, multicast algorithms have become more important due to the speciffc natureof data transmitted in transport networks. Current research work on telecommunication networks considers especially real time data transmission. The increase in traffic capacity of present-day networks has offered great advantages in distributed applications such as multimedia data transmission in real time, video-on-demand, teleconferencing etc.

The implementation of multicasting requires solutions to many combinatorial problems accompanying the construction of optimal transmission trees. In the optimization process one can distinguish: MST (*Minimum Steiner Tree*), and the tree with the shortest paths between the source node and each of the destination nodes SPT (*Shortest Path Tree*). Finding the MST, which is a $\mathcal{NP}$–complete problem, results in a structure with a minimum total cost [1]. Relevant literature provides a wide range of heuristics solving this problem in polynomial time [2], [6], [7].

From the point of view of the application in data transmission, the most commonly used is the KMB algorithm [2]. Other methods minimize the cost of each of the paths between the sender and each of the members of the multicast group by forming a tree from the paths having the least costs. Doar and Leslie [8] show that the total cost of MST constructed by the KMB heuristic is, on average, 5% worse as compared to the exact cost incurred by the MST algorithm [9].

The analysis of the effectiveness of the algorithms known to the authors and the design of the new solutions utilize the numerical simulation based on the abstract model of the existing network. These, in turn, need structures (network models) that reflect most accurately the Internet network.

In modelling the topology of the Internet network, it is not necessary, or even advisable, to describe the whole of the network. The dynamics of the changes of the topology depends on random connections and disconnections of the hosts and does not allow for building a model reflecting a given current structure. From the point of view of the effectiveness of the algorithms under scrutiny, the use of such a approach in the simulation process is not economical and introduces a great complexity of the calculations. An investigation into traffic in particular domains (or autonomous system) as well as into inter-domain traffic is usually sufficient enough because it takes into consideration the majority of events taking place in the whole of the network.

If the communication network is presented as a graph, the result of the implementation of the routing algorithm will be a spanning tree rooted in the source node and including all destination nodes in the multicast group. Two kinds of trees can be distinguished in the process of optimization: MST – *Minimum Steiner Tree*, and the tree with the shortest paths between the source node and each of the destination nodes – SPT (*Shortest Path Tree*). Finding the MST, which is a $\mathcal{NP}$-complete problem, effects in a structure with a minimal total cost. The relevant literature provides a wide range of heuristics solving the above problem in polynomial time [2], [3], [4]. From the point of view of the application in data transmission, the most commonly used is the KMB algorithm [2]. The other method minimizes the cost of each of the paths between the sender and each of the members of the multicast group by forming a tree from the paths having the least costs. Conventionally, it is first either the Dijkstra algorithm [12] or the Bellman-Ford algorithm [5] that is used, and then the branches of the tree that do not have destination nodes are cut off.

The remarks presented above indicate the direction of the research work carried out by the authors. The main goal of these investigations is to elaborate a methodology for a reliable comparison of existing solutions and a proposition of a new algorithm [28]. This methodology should define a wide range of network topologies as a base for the simulation process of multicast algorithms. A set of important parameters that describes networks should be applied as well. Some statistical properties of the results of multicast algorithms are examined in the article.

The article is divided into seven sections. Section 2 describes the implemented network model. Section 3 presents an overview of the STA algorithm. In Section 4, multicast group members distribution methods are laid down. In Section 5, the simulation methodology is described. Section 6 includes the results of the simulation of the implemented algorithms (STA and others), while Section 7 sums up the presented study.

II. Network model

Let us assume that a network is represented by an undirected, connected graph $G = (V, E)$, where V is a set of nodes, and E is a set of links. The existence of the link $e = (u, v)$ between the nodes u and v entails the existence of the link $e' = (v, u)$ for any $u, v \in V$ (corresponding to two-way links in communication networks). With each link $e \in E$, the cost $c(e)$ parameter is coupled. The cost of a connection represents the usage of the link resources. The multicast group is a set of nodes that are receivers of the group traffic (identification is carried out according to a unique i address), $M = \{m_1, ..., m_m\} \subseteq V$, where $m = |M| \leq |V|$. The node $s \in V$ is the source for the multicast group M. Multicast tree $T(s, M) \subseteq E$ is a tree rooted in the source node s that includes all members of the group M and is called a *Steiner tree*. The total cost of the tree $T(s, M)$ can be defined as $\sum_{t \in T(s,M)} C(t)$. The path $P(s, m_i) \subseteq T(s, M)$ is a set of links between s and $m_i \in M$. The cost of path $P(s, m_i)$ can be expressed as: $\sum_{p \in P(s,m_i)} C(p)$.

A Steiner tree is a good representation for solving the routing multicast problem. This approach becomes particularly important when we have to deal with only one active multicast group and the cost of the whole group has to be minimum. However, due to the computational complexity of this algorithm ($\mathcal{NP}$-complete problem) [11], heuristic algorithms are most preferable. If the set of the nodes of the minimum Steiner tree includes all nodes of a given network, then the problem comes down to finding the minimum spanning tree (this solution can be obtained in polynomial time).

III. Overview of the Algorithms

The simplest way of running the routing algorithm for multicast connections is the implementation of one of the classic algorithms constructing a minimum spanning tree, i.e. the Kruskal algorithm [22], or Prim's algorithm [23]. The designated spanning tree is also constructed for $n \neq M$, which, in practice, effects in aggravation of unnecessary traffic in the network since routers must determine paths for each of the nodes. The cost of tree is disproportionately high in relation to results returned by the exact algorithm (MST).

These inconveniences can be solved by the pruning mechanism that can be introduced to the resulting spanning tree. The PPH technique (*Pruned Prim Heuristic*) [19] is a modification of the classical Prim's algorithm which is a good and efficient solution for solving the Steiner problem when $m \approx n$. PPH builds a minimum spanning tree in the network represented by an undirected graph and removes unwanted arcs – branches that do not contain multicast nodes. Our analysis of algorithms results shows that PPH can construct multicast trees with lower costs as compared to the results of the popular SPT algorithm when the *group density* parameter [13] is greater than 0.5. The mode of operation of the PPH algorithm is presented in Algorithm 1.

Algorithm 1 Pruned Prim Heuristic

```
PPH($C$, $s$, $M$)
    $C$ – adjacency matrix with costs of links in graph,
    $s$ – source node,
    $M$ – set of multicast nodes $m_i \in M$.
$T_{full} \leftarrow$ Prim($C, s$)
$T \leftarrow$ DeleteLeaves($T_{full}, M$)
return $T$
```

Kou, Markovsky and Bermann have proposed the following heuristic algorithm (KMB) determining (constructing) a minimum multicast tree [2]:

- for any cohesive, undirected graph $N = (V, E)$ that includes a set of receiving nodes G, construct a cohesive, undirected graph $N_1 = (V_1, E_1)$ that consists of a sending node s only and of a set of receiving nodes G (the paths between the nodes of graph N_1 are the least cost paths in the original graph N),
- determine a minimum spanning tree T_1 for graph G_1 (if there are more than one solution, choose just one),
- construct a subgraph G_S of graph G by replacing each edge of the tree T_1 with a corresponding path from graph G,
- determine a minimum spanning tree T_S for graph G_S (if there are more then one, choose one),
- construct a Steiner tree T_{KMB} form the tree T_S by removing leaves that do not include receiving nodes.

A good representative for the class of algorithms that construct a multicast tree with the shortest paths is the SPT algorithm (*Shortest Path Tree*). The mode of operation is based on constructing the shortest paths tree only for those nodes that are members of the multicast group (Algorithm 2).

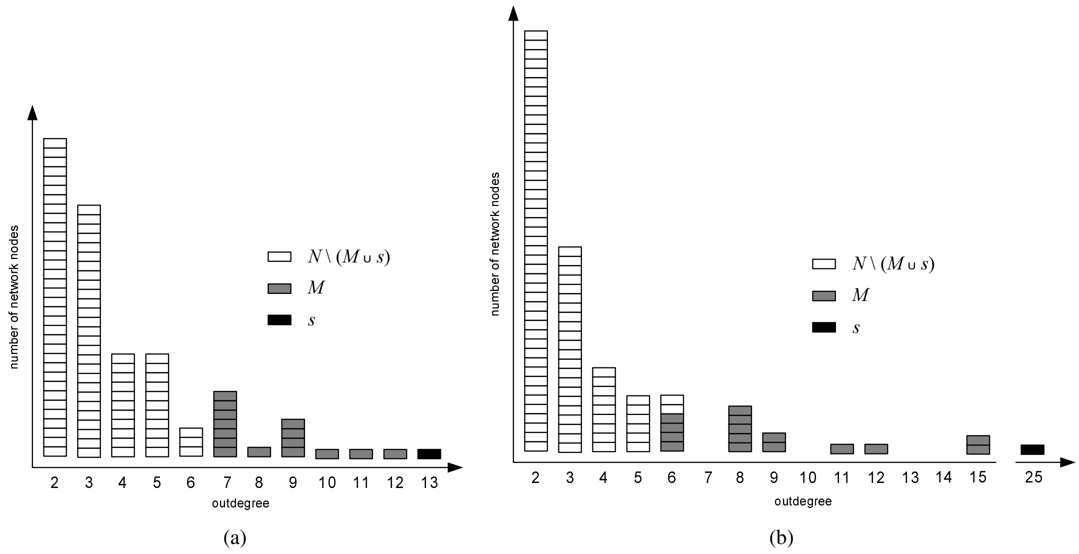


Figure 1. Histogram of the node outdegree distributions and the illustration of the operation of the algorithm *GroupHighDegree* for an exemplary Waxman (a) and Barabási-Albert network (b) ($n = 100$, $m = 15$, $D_{av} = 4$)

Algorithm 2 Shortest Path Tree

```
SPT($C$, $s$, $M$)
  $C$ – adjacency matrix with costs of links in graph,
  $s$ – source node,
  $M$ – set of multicast nodes $m_i \in M$.
for each vertex $m_i \in M$ do
    $p_i \leftarrow$ Dijkstra($C, s, m_i$)
    AddPath($p_i, T$)
end for
DeleteLeaves($T, M$)
return $T$
```

Algorithm 3 Switched Tree Algorithm

```
STA($C$, $s$, $M$)
  $C$ – adjacency matrix with costs of links in graph,
  $s$ – source node,
  $M$ – set of multicast nodes $m_i \in M$.
$T_{SPT} \leftarrow$ SPT($C$, $s$, $M$)
$T_{PPH} \leftarrow$ PPH($C$, $s$, $M$)
if $c_{SPT} > c_{PPH}$ then
    $T := T_{PPH}$
else
    $T := T_{STA}$
end if
return $T$
```

The above observation makes its possible to propose a decision mechanism (STA) which chooses a multicast tree with minimum cost obtained from SPT or PPH that can work concurrently (Algorithm 3). The STA (*Switched Tree Algorithm*) mechanism is based on two well-known optimization algorithms: SPT (*Shortest Path Tree*) and PPH (*Pruned Prim Heuristic*). The combination of these two solutions allows us to achieve a better performance than with the case of each of them working separately. The main concept of the SPT algorithm is to build a shortest paths tree between the source node (s) and each of multicast nodes (m_i) using the Dijkstra algorithm [12]. In the last step of SPT, loops in graphs are removed using the Prim's algorithm and nodes with outdegree 1 which are not multicast members are pruned as well. The STA technique is easy to implement and very fast.

IV. Multicast group members distribution

An essential element of the conducted research study process is a determination of methods for the distribution of the multicast group members in a network [28]. Having in mind various implementations of methods for generating many network topologies, there is also a need for a wider mechanism determining receiving nodes in a network by geographical positioning or one that would be related to the link and node parameters (for example, the outdegree of the network). This will allow us to answer the question whether the way receiving nodes are distributed has any influence on the quality of multicast trees constructed by algorithms. The following methods were used in the study:

- *GroupRandom* method,
  A group of receiving nodes M is constructed by a random choice of m network nodes from among all its nodes (N). The source node s is also randomly chosen from among the number n of nodes in the network. Apparently, this is the only method that has been used so far in any research studies [14], [15], [16], [17], [18].

- *GroupRadius* method,
  Nodes that form a multicast group M and the source node s are chosen from N_r nodes within a circle with radius $r = \frac{d_m}{2}$. This method for creating a group was presented by the authors in [19]. The method reflects the geographical distribution of nodes in a real network.
- *GroupHighDegree* method.
  The algorithm used for the purpose determines the *outdegree* of the network – the number of outcoming links from each node of the network $i \in N$ – and then sorts out the nodes in the diminishing order of this outdegree value. The group $\{M \cup s\}$ is constructed from $m+1$ of the most-preferred nodes (with the highest number of links). Figure 1 shows a histogram for the node outdegree distributions in exemplary (sample) networks generated by the Waxman and the Barabási-Albert methods and explains the operation of the algorithm *GroupHighDegree*.

The method for the receiving nodes M distribution in the network has not been addressed and analyzed earlier in literature.

V. Network topology

A. Generative methods

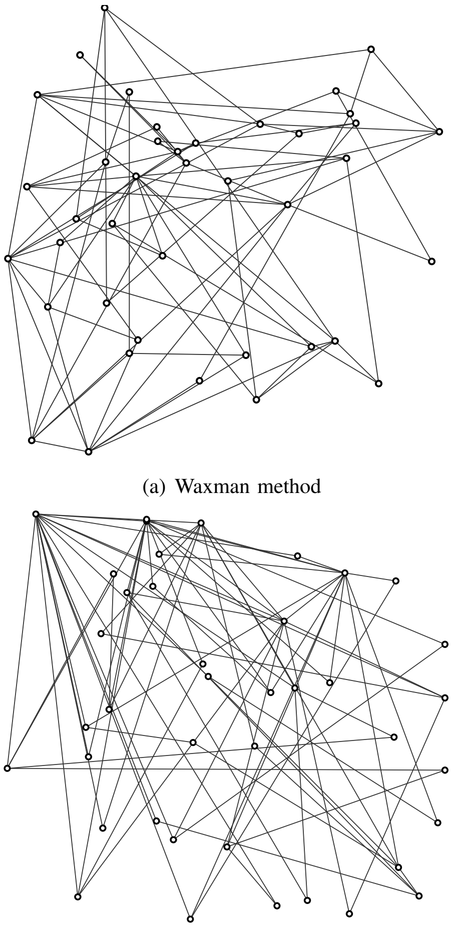
The Internet is a set of nodes interconnected with links. This simple definition makes it possible to represent this real structure as a graph. In fact, the Internet is a set of domains – a number of grouped nodes (routers) which are under joined administration and share routing information. The Internet consists of thousands of domains and autonomous systems (AS). It is possible to generate those kinds of synthetic structures reflecting real topologies [20].

In the study, a flat random graph constructed according to the Waxman method was used [1]. This method defines the probability of an edge between node u and v as:

$$P(u,v) = \alpha e^{\frac{-d}{\beta L}} \tag{1}$$

where $0 < \alpha, \beta \leq 1$, d is an Euclidean distance between the node u and v, and $L = \sqrt{2}$ is the maximum distance between two freely selected nodes. An increase in the parameter α effects in the increase in the number of edges in the graph, while a decrease of the parameter β increases the ratio of the long edges against the short ones.

Another method was proposed by Barabasi in [21]. This model suggests two possible causes for the emergence of a power law in the frequency of outdegrees in network topologies: incremental growth and preferential connectivity. The network growth process consist of incremental addition of new nodes. Preferential connectivity refers to the tendency of a new node to connect to existing nodes that are highly connected or popular. When a node u connects to the network, the probability that it connects to a node v (already belonging to the network) is yielded by:



(a) Waxman method

(b) Barabasi method

Figure 2. Visualization of network topologies (D_{av}=4, n=40)

$$P(u,v) = \frac{d_v}{\sum_{k \in V} d_k} \tag{2}$$

where d_v is the degree of a node belonging to the network, V is the set of nodes connected to the network, and $\sum_{k \in V} d_k$ is the sum of the outdegrees of the nodes previously connected.

With the construction of the network models based on Waxman and Barabasi-Albert method, BRITE (*Boston university Representative Internet Topology gEnerator*) [27] was used as a tool for generation of realistic topologies. The application provides a range of network topology models and appropriate generative methods.

Fig. 2 shows typical topologies generated with the application of the Waxman and Barabasi-Albert method.

A network model was adopted in which the nodes were arranged on a square grid with the size of 1000×1000 (Waxman parameters: $\alpha = 0.15$, $\beta = 0.2$). Onto the existing network of connections, the cost matrix $C(u,v)$ was applied (as a adjacency matrix of Euclidean distances between the nodes).

It was an important element during the simulation process to maintain a steady average node degree of the graph (for each of the generated networks) defined as: $D_{av} = \frac{2k}{n}$ (where n is the number of the nodes of the network, k is the number of the edges) which, in practice, meant the necessity of maintaining a steady number of edges. In the implementations, the adopted degree of the graph was 4.

B. Parameters

The efficiency of multicast algorithms depends on the implemented network structure. Thus, it is important to define the basic parameters that describe the network topology:

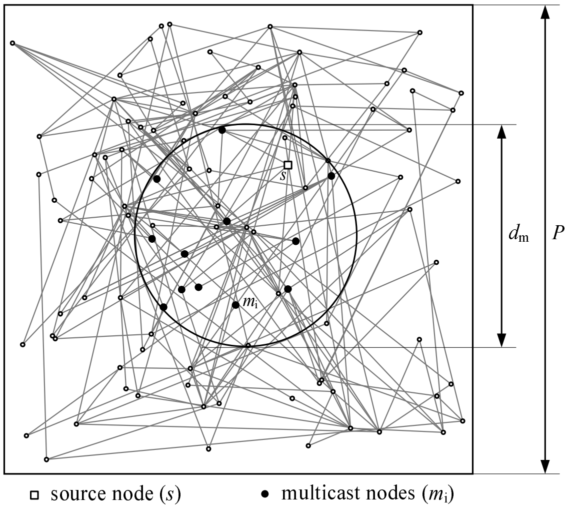


Figure 3. The explanation of the idea of the group spreading factor ε_m ($n = 200,\ D_{av} = 4$)

- *hop diameter* - is the length of the longest shortest-path between any two nodes; shortest paths are computed using *hop count* metric,
- *length diameter* - is the length of the longest shortest-path between any two nodes; shortest paths are computed using the Euclidean distance metric,
- *clustering coefficient* (γ_v) of node v is the proportion of links between the vertices within its neighbourhood divided by the number of links that could possibly exist between them [24].

  Let $\Gamma(v)$ be a neighborhood of a vertex v consisting of the vertices adjacent to v (not including v itself). More precisely:

$$\gamma_v = \frac{|E(\Gamma(v))|}{\binom{k_v}{2}} = \frac{|E(\Gamma(v))|}{k_v(k_v-1)} \tag{3}$$

  where $|E(\Gamma(v))|$ is the number of edges in the neighborhood of v and $\binom{k_v}{2}$ is the total number of possible edges between neighborhood nodes.

  Let $V^{(1)} \subset V$ denote the set of vertices of degree 1. Therefore [25], [26]:

$$\hat{\gamma} = \frac{1}{|V| - |V^{(1)}|} \sum_{v \in V} \gamma_v \tag{4}$$

- *group spreading factor* (ε_m) – describes the arrangement of the multicast group on the plane (Fig. 3). It is defined as the diameter of the area (d_m) containing all multicast nodes divided by the size of plane (P) where all the nodes are situated (1).

$$\varepsilon_m = \frac{d_m}{P}, \tag{5}$$

VI. Numerical results

In the study, a flat random graph constructed according to the Waxman [1] and Barabási [21] methods was used to generate networks topologies to validate the accuracy of our algorithm. With the construction of the network models based on the Waxman and the Barabási-Albert methods, BRITE was used as a tool for generating realistic topologies. The application provides a range of network topology models and appropriate generative methods. The research work was conducted with the application of the networks generated by the above-mentioned methods that were appropriately adopted and unified [29], [30], [31], [32], [33], [35].

The numerical results are divided into three stages. The first stage of the experiment investigates the efficiency of STA and other algorithms. The evaluation of the new constrained algorithm STA and the existing solutions (KMB, DDMC and SPT) bases on the average cost of multicast trees.

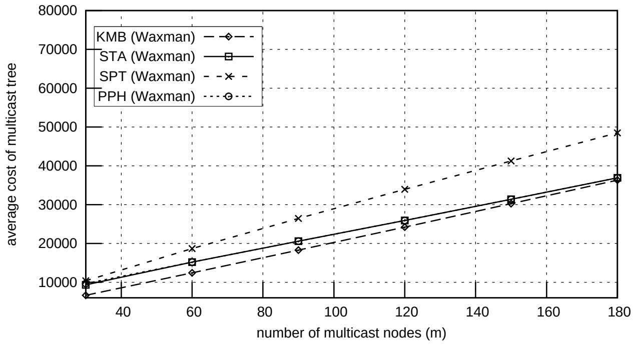


Figure 4. Average cost of multicast tree versus the number of multicast nodes m for Waxman model ($n = 200,\ D_{av} = 4$)

The quality of STA can be observed in Figures 4 and 5. Regarding fixed network parameters, STA bases mainly on PPH heuristic that constructs trees with lower costs as compared to SPT. The convergence of the STA and the KMB results for large groups occurs in the two examined network topology models. The multicast group number growth causes

Table I
VALUE OF THE AVERAGE COST OF MULTICAST TREE CONSTRUCTED BY THE ALGORITHMS WITHOUT CONSTRAINTS FOR DIFFERENT METHODS FOR DISTRIBUTION OF RECEIVING NODES AND METHODS FOR MODELING NETWORK TOPOLOGY ($n = 100$, $m = 20$, $D_{av} = 4$)

model	multicast group	KMB	DDMC	STA	SPT	PPH
Waxman	*GroupRandom*	4732	4739	6442	7252	6651
	GroupRadius$(0, 2)$	5594	5742	6634	6962	7156
	GroupRadius$(0, 4)$	4836	4867	6200	6655	6565
	GroupHighDegree	4797	4819	6294	7230	6427
Barabási	*GroupRandom*	7241	7243	8959	9618	9405
	GroupRadius$(0, 2)$	9058	9463	9902	10273	10409
	GroupRadius$(0, 4)$	7590	7721	8887	9316	9435
	GroupHighDegree	7275	7282	8977	10030	9240

Table II
VALUE OF THE AVERAGE COST OF MULTICAST TREE CONSTRUCTED BY THE ALGORITHMS WITHOUT CONSTRAINTS FOR DIFFERENT METHODS OF DISTRIBUTION OF RECEIVING NODES AND METHODS FOR MODELING NETWORK TOPOLOGY ($n = 500$, $m = 100$, $D_{av} = 4$)

model	multicast group	KMB	DDMC	STA	SPT	PPH
Waxman	*GroupRandom*	19998	20022	25814	30286	25854
	GroupRadius$(0, 2)$	28631	29386	34212	36189	34493
	GroupRadius$(0, 4)$	21909	22102	27554	30523	27692
	GroupHighDegree	20697	20786	25819	30710	25827
Barabási	*GroupRandom*	35230	35243	42489	45979	42777
	GroupRadius$(0, 2)$	43884	45891	47781	50420	48006
	GroupRadius$(0, 4)$	36344	36928	42161	44987	42434
	GroupHighDegree	35468	35505	42470	48175	42533

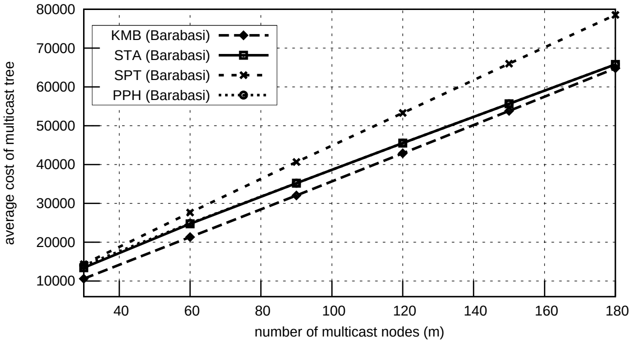


Figure 5. Average cost of multicast tree versus the number of multicast nodes m for Barabási-Albert model ($n = 200$, $D_{av} = 4$)

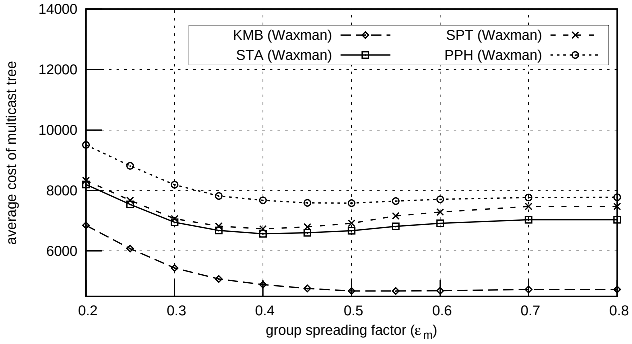


Figure 6. Average cost of multicast tree versus the group spreading factor ϵ_m for Waxman model ($n = 200$, $m = 20$, $D_{av} = 4$)

a growth in the average cost of multicast trees. The costs increase linearly.

Beside conventional parameters of simulation, such as the number of multicast nodes in the network m, we also took into consideration the group spreading factor (ε_m) [19]. A change in this parameter influences the way receiving nodes are distributed in the implemented network. The influence of the group spreading factor ε_m on the average cost of the multicast tree was examined for fixed values of the network parameters,(Figures 6 and 7).

In order to choose and mark nodes in the networks as group members in the area bounded by a circle, the *GroupRadius* method was implemented. The number of network nodes $m = 15$ is required to constitute the multicast group in a narrow area ($\varepsilon_m = 0.2$).

The dependency of group spreading factor is observable for all the examined algorithms. An increase in the area bounded by the circle is followed by a decrease in the average costs of trees constructed by the KMB algorithm. To be more precise, decreasing the factor ε_m to the value 0.2 effects in an increase in the costs of trees 60% on average for the Waxman model (Fig. 6) and 48% for the Barabási-Albert model (Fig. 7). The proposed STA algorithm is less sensitive – the difference is 27%. Above the value $\varepsilon_m = 0.7$, multicast nodes spread randomly throughout the whole network and the costs of trees are similar to those obtained by the

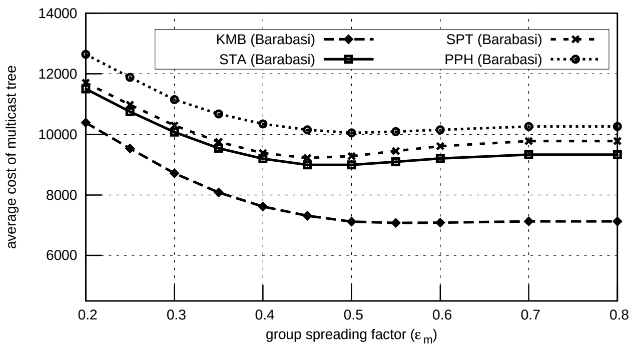


Figure 7. Average cost of multicast tree versus the group spreading factor ϵ_m for Barabási-Albert model ($n = 200$, $m = 20$, $D_{av} = 4$)

GroupRandom method.

Using the methods proposed in Section IV, a research study on the influence of these methods on the average costs of multicast trees was carried out in relation to the following parameters:

- method of generating network topology (Waxman or Barabási-Albert methods),
- method for estimation of the costs of links (as the Euclidean distance and randomly chosen from the interval $10 \dots 1,000$),
- change in the scale (linear increase of the number of network nodes and receiving nodes) – for ($n = 100$, $m = 20$) and ($n = 500$, $m = 100$).

The indicated studies make sense mainly for algorithms without constraints, where the cost metric is the Euclidean distance, thus being heavily dependant on the distribution of nodes in the plane. The results of the studies carried out for link costs represented as Euclidean metrics are presented in Tables I and II.

Basing on observations related to the influence of the parameter ε_m on the costs of obtained trees, the characteristic values $\varepsilon_m = 0.2$ and $\varepsilon_m = 0.4$ were chosen, for which the above-mentioned costs were respectively the highest and the lowest [19].

The analysis of the average costs of trees indicates the existence of the influence of one particular method applied in the distribution of receiving nodes in the network. Depending on the algorithm under scrutiny, the lowest costs of trees are obtained for the *GroupRandom* method and the *GroupHighDegree* method. The differences in the results for the same algorithm in relation to the applied method fluctuate within the interval 7–21% (Table I, Waxman model), whereas the DDMC algorithm is the most sensitive to the way multicast nodes are distributed (21%), and the least sensitive is the STA algorithm (7%). It should be also noted that these differences are maintained at a similar level for the Barabási-Albert model (10–30%).

Reliability of conducted studies requires an experimental phase that would take into consideration other network parameters. For this purpose, the number of nodes in the network was increased fivefold ($n = 500$) and the number of multicast nodes to ($m = 100$). Similarly to the previous case, the lowest costs of trees were obtained for the *GroupRandom* method and for the *GroupHighDegree* method. The differences in the results for the same algorithm in relation to the applied method fluctuate between 19–46% for the Waxman model (Table II). The most sensitive for the method of multicast nodes distribution is the DDMC algorithm (46%), while the least sensitive is the SPT algorithm (19%). For the Barabási-Albert model, the results are the lowest (10–30%).

The conclusions of thus conducted study, however, are not so conclusive and obvious. There is, indeed, a dependency between the algorithms and the methods for distribution of receiving nodes, but every analyzed algorithm requires, however, an individual approach, with such criteria taken into consideration as: the result of the algorithm (cost of tree or path), the model used and the network parameters, the number of receiving nodes, and so on. For example, the lowest costs of tree are yielded by the DDMC algorithm in networks with 500 nodes and 100 multicast nodes generated by the Waxman model with the application of the *GroupRandom* method as compared with the *GroupRadius*(0.2) method (Table II).

The authors made numerous experiments for different topologies to develop appropriate research methodology. The research investigation was conducted for 1,000 networks.

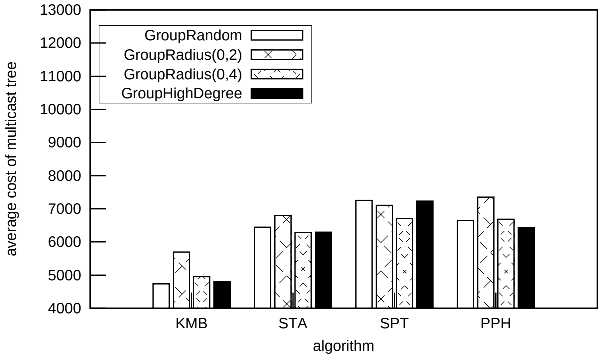


Figure 8. Value of the average cost of multicast tree for different methods for distribution of receiving nodes and Waxman model ($n = 100$, $m = 20$, $D_{av} = 4$)

A similar analysis was made for a cost metric that was a random value taken from within the interval $10 \dots 1000$ (Figs. 8 and 9). The findings show that the algorithms optimizing cost of trees (KMB and PPH) are more effective with the application of the *GroupRandom* and the *GroupHighDegree* methods. In turn, the SPT algorithm yields the lowest costs of trees for the *GroupRandom* method and the *GroupRadius* method with the parameter $\varepsilon_m = 0.2$. The above observations are correct and accurate both for the

Table III
DESCRIPTIVE STATISTICS OF UNCONSTRAINED ALGORITHMS RESULTS FOR 1,000 NETWORKS ($n = 200$, $m = 20$, $D_{av} = 4$)

model	parameter	KMB	DDMC	STA	SPT	PPH
Waxman	mean value	4731,2	4741,0	7037,8	7471,7	7774,0
	minimum value	3275,0	3275,0	4268,0	4473,0	4268,0
	maximum value	6436,0	6569,0	10385,0	11817,0	12752,0
	standard deviation	527,1	531,9	1012,2	1189,3	1309,0
	variation coefficient	0,111	0,112	0,143	0,159	0,168
	skewness	0,251	0,241	0,227	0,368	0,262
	kurtosis	0,056	0,033	-0,037	-0,038	0,001
Barabási	mean value	7130,1	7134,1	9328,8	9779,9	10260,0
	minimum value	4607,0	4607,0	5673,0	5863,0	5673,0
	maximum value	10616,0	10616,0	16689,0	19314,0	16689,0
	standard deviation	910,1	914,9	1525,2	1690,3	1820,6
	variation coefficient	0,127	0,128	0,163	0,172	0,177
	skewness	0,388	0,398	0,506	0,627	0,374
	kurtosis	0,323	0,331	0,479	0,943	-0,084

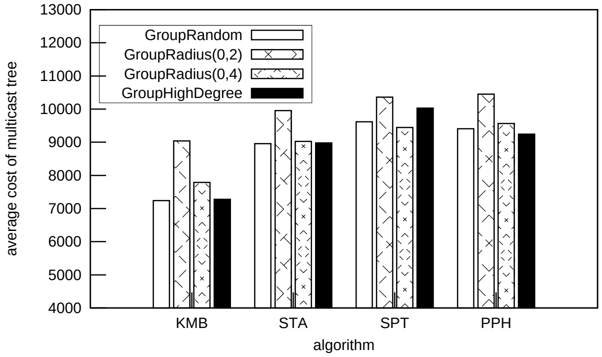


Figure 9. Value of the average cost of multicast tree for different methods for distribution of receiving nodes and Barabási-Albert model ($n = 100$, $m = 20$, $D_{av} = 4$)

Waxman method and for the Barabási-Albert method. The differences in the costs of trees within the same algorithm are 8–30%.

The third stage of the research investigation was conducted for 1,000 networks and the costs of the obtained trees were averaged. The implementation of descriptive statistics as standard deviation, minimum and maximum value, variation coefficient, skewness and kurtosis, allows to confirm the efficacy of this approach (Table III).

The standard deviation is the lowest for the KMB and the DDMC algorithms. It has a slightly higher value for the networks generated with the application of the Barabási-Albert model. The values of the variation coefficient show that the results of KMB, DDMC and STA are least differentiated. The analysis of the skewness parameter indicates an asymmetry (positive skew) of the distribution of the algorithms results. The asymmetry is maximum for the STA algorithm. In the case of the Waxman model, the distributions of algorithms results are close to a normal distribution. The implementation of the Barabási-Albert model leads to a leptokurtic distribution (similar to gamma distribution).

Kurtosis defines a relative measure of the concentration and the flatness of the dispersion of the results for costs of trees. In the case of the Waxman model, the result dispersion has a shape similar to normal dispersion (mesokurtic). This observation applies to all algorithms under scrutiny in the present study. The implementation of the Barabási-Albert model is followed by large values of kurtosis – the dispersion graph becomes more slender than that for normal dispersion (except for the PPH algorithm).

A comparison of the algorithm requires a construction of confidence intervals for average values. The construction of confidence intervals makes it possible to examine whether it is correct to compare the algorithms results only on the basis of the average obtained from 1,000 results. The data presented in Table IV are arranged according to the ascending value of the average cost of trees c_T. The analysis of the ordered data in the table allows us to observe that also the edges of the confidence intervals for both generative methods are arranged in the ascending order. Moreover, except the KMB and DDMC algorithms – for which the values are comparable, the confidence intervals of the remaining algorithms do not overlap. For the given network parameters and algorithm parameters, the following dependence is then correct:

$$c_{KMB} \approx c_{DDMC} < c_{STA} < c_{SPT} < c_{PPH}. \qquad (6)$$

VII. CONCLUSIONS

The article proposes a novel multicast routing algorithm without constraints and introduces the group members arrangement as a new parameter for analyzing multicast routing algorithms finding multicast trees. In the article, the results of the proposed STA algorithm are compared with the representative algorithms without constrains.

The article extends the existing methodology for the evaluation of multicast routing algorithms. Analyzing group

Table IV
CONFIDENCE INTERVALS FOR AVERAGE VALUES OF COSTS OF TREES FOR GROUP TRANSMISSION CONSTRUCTED BY STUDIED ALGORITHMS WITHOUT CONSTRAINT FOR 1,000 NETWORKS ($n = 200$, $m = 20$, $D_{av} = 4$)

algorithm	Waxman model			Barabási-Albert model		
	$\overline{c_T} - u_\alpha \frac{S_N}{\sqrt{N}}$	$\overline{c_T}$	$\overline{c_T} + u_\alpha \frac{S_N}{\sqrt{N}}$	$\overline{c_T} - u_\alpha \frac{S_N}{\sqrt{N}}$	$\overline{c_T}$	$\overline{c_T} + u_\alpha \frac{S_N}{\sqrt{N}}$
KMB	4698,53	4731,20	4763,88	7073,68	7130,08	7186,48
DDMC	4708,04	4741,01	4773,98	7077,45	7134,10	7190,75
STA	6975,04	7037,78	7100,51	9234,30	9328,84	9423,37
SPT	7397,94	7471,65	7545,37	9675,09	9779,85	9884,62
PPH	7692,86	7773,99	7855,12	10147,39	10260,24	10373,08

members arrangement constitutes an essential input in the research methodology. This kind of approach has not been considered in relevant literature so far. The article examines unconstrained routing algorithms for multicast connections emphasizing the quality of the network model (the accuracy of the illustration of a real Internet topology) and presents a new proposal.

The research has been conducted by the authors for several years. Initially, the studies focused on the accuracy of multicast routing algorithms in relation to the exact algorithm (MST) and were provided for networks consisting of several nodes [34]. The next stage of research work evaluated the inffuence of the parameters describing graphs (that represents real topologies) on the costs of trees constructed by examined algorithms [29], [30]. The studies are unique because they analyze the algorithms in wide range of network sizes (from several to ten thousand nodes) [31], [33]. Analyses are conducted for networks generated as random graphs with an implementation of Waxman and Barabási-Albert method and with an application of Inet heuristic generator. Separate track of the research analyzed the inffuence of network topology parameters on the cost of multicast tree constructed by selected genetic algorithms [35].

The research results show that obtaining of a tree with the lower cost can be the result of the application of the generative methods and not only the result of the application of a more efficient routing algorithm. Although the Barabási-Albert method returns trees with higher costs, it reflects the real topology of the Internet in the most accurate way. The study also shows that the selection of the generative method should depend on the size of the network (the number of the nodes) and the size of the multicast group.

Finally, authors have proposed the simulation methodology that reflect some network parameters proposed by authors and examine these parameters influence on the costs of multicast trees constructed by multicast routing algorithms.

REFERENCES

[1] B. Waxmann, “Routing of multipoint connections,” *IEEE Journal on Selected Area in Communications*, vol. 6, pp. 1617–1622, 1988.

[2] L. Kou, G. Markowsky, and L. Berman, “A fast algorithm for Steiner trees,” *Acta Informatica*, no. 15, pp. 141–145, 1981.

[3] J. S. Crawford and A.G. Waters, “Heuristics for ATM Multicast Routing,” *Computer Science*, University of Kent at Canterbury, 1998.

[4] Q. Sun and H. Langendoerfer, “Efficient Multicast Routing for Delay-Sensitive Applications,” *Proceedings of the 2-nd Workshop on Protocols for Multimedia Systems (PROMS’95)*, October 1995, pp. 452–458.

[5] R. Bellman, “On a routing problem,” *Quarterly of Applied Mathematics*, vol. 16, no. 1, pp. 87–90, 1958.

[6] Q. Zhu, M. Parsa, and J. J. Garcia-Luna-Aceves, “A source-based algorithm for delay-constrained minimum-cost multicasting,” in *INFOCOM ’95: Proceedings of the Fourteenth Annual Joint Conference of the IEEE Computer and Communication Societies (Vol. 1)-Volume*. IEEE Computer Society, 1995, p. 377.

[7] A. Shaikh and K. G. Shin, “Destination-Driven Routing for Low-Cost Multicast.” *IEEE Journal on Selected Areas in Communications*, vol. 15, no. 3, pp. 373–381, 1997.

[8] M. Doar and I. Leslie, “How bad is naive multicast routing?” *IEEE INFOCOM’ 1993*, pp. 82–89, 1993.

[9] S. Hakimi, “Steiner’s problem in graphs and its implications,” *Networks*, vol. 1, pp. 113–133, 1971.

[10] E. W. Zegura, K. L. Calvert, and S. Bhattacharjee, “How to Model an Internetwork,” *IEEE INFOCOM ’96*, 1996.

[11] R. Karp, “Reducibility among combinatorial problems,” *Complexity of Computer Computations*, pp. 85–104, 1972.

[12] E. Dijkstra, “A note on two problems in connexion with graphs,” *Numerische Mathematik*, vol. 1, pp. 269–271, 1959.

[13] F. Cheng and R. Chang, “A tree switching protocol for multicast state reduction,” in *IEEE Symposium on Computers and Communications ISCC 2000*, 2000, pp. 672–677.

[14] L. Wei and D. Estrin, "The trade-offs of multicast trees and algorithms," in *Proceedings of ICCCN'94*. IEEE, 1994, pp. 55–64.

[15] G. Rouskas and I. Baldine, "Multicast Routing with End-to-End Delay and Delay Variation Constraints," *IEEE Journal on Selected Areas in Communications*, no. 15, pp. 346–356, 1997.

[16] G. Liu and K. Ramakrishnan, "A*Prune: An Algorithm for Finding K Shortest Paths Subject to Multiple Constraints," in *INFOCOM 2001 Twentieth Annual Joint Conference of the IEEE Computer and Communications Societies*, 2001, pp. 743–749.

[17] E. Gelenbe, A. Ghanwani, and V. Srinivasan, "Improved neural heuristics for multicast routing," *IEEE Journal on Selected Areas in Communications*, no. 15, pp. 147–155, 1997.

[18] D. Thaler and C. Ravishankar, "Distributed center-location algorithms," *IEEE Journal on Selected Areas in Communications*, no. 15, pp. 291–303, 1997.

[19] M. Piechowiak, P. Zwierzykowski, and T. Bartczak, "An Application of the Switched Tree Mechanism in the Multicast Routing Algorithms," in *1-st Interdisciplinary Technical Conference of Young Scientists InterTech 2008*, 2008, pp. 282–286 (the best paper award).

[20] M. Newman, A. Barabási, and D. Watts, Eds., *The Structure and Dynamics of Networks*, ser. Princeton Studies in Complexity. Princeton University Press, 2007.

[21] A. L. Barabasi and R. Albert, "Emergence of scaling in random networks," *Science*, pp. 509–512, 1999.

[22] R. Kruskal, "Mininum Spanning Tree," *Proceedings of the American Mathematical Society*, pp. 48–50, 1956.

[23] R. Prim, "Shortest connection networks and some generalizations," *Bell Systems Tech J.*, vol. 36, 1957.

[24] D. J. Watts and S. H. Strogatz, "Collective dynamics of 'smallworld' networks," *Nature*, vol. 12, no. 393, pp. 440–442, 1998.

[25] M. Faloutsos, P. Faloutsos, and C. Faloutsos, "On Power-Law Relationships of the Internet Topology," *ACM Computer Communication Review, Cambridge, MA*, pp. 111-122, 1999.

[26] T. Bu and D. Towsley, "On distinguishing between Internet power law topology generators," *Proceedings of INFOCOM 2002*, 2002.

[27] A. Medina, A. Lakhina, I. Matta, and J.Byers, "BRITE: An Approach to Universal Topology Generation," *IEEE/ACM MASCOTS*, pp. 346–356, 2001.

[28] M. Piechowiak and M. Stasiak and P. Zwierzykowski, "Analysis of the Influence of Group Members Arrangement on the Multicast Tree Cost," in *Proceedings of The 5-th Advanced International Conference on Telecommunications*, Venice, Italy, May 2009 (the best paper award).

[29] M. Piechowiak and P. Zwierzykowski, "The Influence of Network Topology on the Efficiency of Multicast Heuristic Algorithms," in *Proceedings of The 5-th International Symposium - Communication Systems, Networks and Digital Signal Processing*, M. Logothetis, Ed., July 2006, pp. 115–119.

[30] M. Piechowiak and P. Zwierzykowski, "The Application of Network Generation Methods in the Study of Multicast Routing Algorithms," in *Proceedings of 4th International Working Conference on Performance Modelling and Evaluation of Heterogeneous Networks HET-NETs 2006*. Ilkley, UK: Networks UK Publishers, September 2006, pp. P24/1–8.

[31] M. Piechowiak and P. Zwierzykowski, "Performance of Fast Multicast Algorithms in Real Networks," in *Proceedings of EUROCON 2007 The International Conference on: Computer as a tool*, IEEE. Warsaw, Poland: IEEE, September 2007, pp. 956–961.

[32] M. Piechowiak and P. Zwierzykowski, "Heuristic Algorithm for Multicast Connections in Packet Networks," in *Proceedings of EUROCON 2007 The International Conference on: Computer as a tool*, IEEE. Warsaw, Poland: IEEE, September 2007, pp. 948–955.

[33] M. Piechowiak and P. Zwierzykowski, "Efficiency Analysis of Multicast Routing Algorithms in Large Networks," in *Proceedings of The Third International Conference on Networking and Services ICNS 2007*, IEEE. Athens, Greece: IEEE, June 2007, pp. 101–106 (the best paper award).

[34] M. Piechowiak and P. Zwierzykowski, "Analiza efektywności algorytmów heurystycznych dla połączeń rozgałęźnych w sieciach pakietowych,"*Krajowe Sympozjum Telekomunikacji*, pp.192-200, September, 2005 (in polish).

[35] A. Chojnacki, M. Piechowiak, and P. Zwierzykowski, "Genetic Routing Algorithm for Multicast Connections in Packet Networks," *International Journal of Image Processing & Communications*, vol. 13, no. 1-2, pp. 13–20, 2008.

www.iariajournals.org

International Journal On Advances in Intelligent Systems
- ICAS, ACHI, ICCGI, UBICOMM, ADVCOMP, CENTRIC, GEOProcessing, SEMAPRO, BIOSYSCOM, BIOINFO, BIOTECHNO, FUTURE COMPUTING, SERVICE COMPUTATION, COGNITIVE, ADAPTIVE, CONTENT, PATTERNS
- issn: 1942-2679

International Journal On Advances in Internet Technology
- ICDS, ICIW, CTRQ, UBICOMM, ICSNC, AFIN, INTERNET, AP2PS, EMERGING
- issn: 1942-2652

International Journal On Advances in Life Sciences
- eTELEMED, eKNOW, eL&mL, BIODIV, BIOENVIRONMENT, BIOGREEN, BIOSYSCOM, BIOINFO, BIOTECHNO
- issn: 1942-2660

International Journal On Advances in Networks and Services
- ICN, ICNS, ICIW, ICWMC, SENSORCOMM, MESH, CENTRIC, MMEDIA, SERVICE COMPUTATION
- issn: 1942-2644

International Journal On Advances in Security
- ICQNM, SECURWARE, MESH, DEPEND, INTERNET, CYBERLAWS
- issn: 1942-2636

International Journal On Advances in Software
- ICSEA, ICCGI, ADVCOMP, GEOProcessing, DBKDA, INTENSIVE, VALID, SIMUL, FUTURE COMPUTING, SERVICE COMPUTATION, COGNITIVE, ADAPTIVE, CONTENT, PATTERNS
- issn: 1942-2628

International Journal On Advances in Systems and Measurements
- ICQNM, ICONS, ICIMP, SENSORCOMM, CENICS, VALID, SIMUL
- issn: 1942-261x

International Journal On Advances in Telecommunications
- AICT, ICDT, ICWMC, ICSNC, CTRQ, SPACOMM, MMEDIA
- issn: 1942-2601