

International Journal on Advances in Intelligent Systems



The *International Journal on Advances in Intelligent Systems* is Published by IARIA.

ISSN: 1942-2679

journals site: <http://www.ariajournals.org>

contact: petre@aria.org

Responsibility for the contents rests upon the authors and not upon IARIA, nor on IARIA volunteers, staff, or contractors.

IARIA is the owner of the publication and of editorial aspects. IARIA reserves the right to update the content for quality improvements.

Abstracting is permitted with credit to the source. Libraries are permitted to photocopy or print, providing the reference is mentioned and that the resulting material is made available at no cost.

Reference should mention:

International Journal on Advances in Intelligent Systems, issn 1942-2679
vol. 15, no. 1 & 2, year 2022, http://www.ariajournals.org/intelligent_systems/

The copyright for each included paper belongs to the authors. Republishing of same material, by authors or persons or organizations, is not allowed. Reprint rights can be granted by IARIA or by the authors, and must include proper reference.

Reference to an article in the journal is as follows:

<Author list>, "<Article title>"
International Journal on Advances in Intelligent Systems, issn 1942-2679
vol. 15, no. 1 & 2, year 2022, <start page>:<end page> , http://www.ariajournals.org/intelligent_systems/

IARIA journals are made available for free, proving the appropriate references are made when their content is used.

Sponsored by IARIA

www.aria.org

Copyright © 2022 IARIA

Editor-in-Chief

Hans-Werner Sehring, Namics AG, Germany

Editorial Advisory Board

Josef Noll, UiO/UNIK, Norway
Filip Zavoral, Charles University Prague, Czech Republic
John Terzakis, Intel, USA
Freimut Bodendorf, University of Erlangen-Nuernberg, Germany
Haibin Liu, China Aerospace Science and Technology Corporation, China
Arne Koschel, Applied University of Sciences and Arts, Hannover, Germany
Malgorzata Pankowska, University of Economics, Poland
Ingo Schwab, University of Applied Sciences Karlsruhe, Germany

Editorial Board

Jemal Abawajy, Deakin University - Victoria, Australia
Sherif Abdelwahed, Mississippi State University, USA
Habtamu Abie, Norwegian Computing Center/Norsk Regnesentral-Blindern, Norway
Siby Abraham, University of Mumbai, India
Witold Abramowicz, Poznan University of Economics, Poland
Imad Abugessaisa, Karolinska Institutet, Sweden
Leila Alem, The Commonwealth Scientific and Industrial Research Organisation (CSIRO), Australia
Panos Alexopoulos, ISOCO, Spain
Vincenzo Ambriola, Università di Pisa, Italy
Junia Anacleto, Federal University of Sao Carlos, Brazil
Razvan Andonie, Central Washington University, USA
Cosimo Anglano, DISIT - Computer Science Institute, Università del Piemonte Orientale, Italy
Richard Anthony, University of Greenwich, UK
Avi Arampatzis, Democritus University of Thrace, Greece
Sofia Athenikos, Flipboard, USA
Isabel Azevedo, ISEP-IPP, Portugal
Ebrahim Bagheri, Athabasca University, Canada
Fernanda Baiao, Federal University of the state of Rio de Janeiro (UNIRIO), Brazil
Flavien Balbo, University of Paris Dauphine, France
Suliman Bani-Ahmad, School of Information Technology, Al-Balqa Applied University, Jordan
Ali Barati, Islamic Azad University, Dezfoul Branch, Iran
Henri Basson, University of Lille North of France (Littoral), France
Carlos Becker Westphall, Federal University of Santa Catarina, Brazil
Petr Berka, University of Economics, Czech Republic
Julita Bermejo-Alonso, Universidad Politécnica de Madrid, Spain
Aurelio Bermúdez Marín, Universidad de Castilla-La Mancha, Spain
Lasse Berntzen, University College of Southeast, Norway
Michela Bertolotto, University College Dublin, Ireland
Ateet Bhalla, Independent Consultant, India
Freimut Bodendorf, Universität Erlangen-Nürnberg, Germany

Karsten Böhm, FH Kufstein Tirol - University of Applied Sciences, Austria
Pierre Borne, Ecole Centrale de Lille, France
Christos Bouras, University of Patras, Greece
Anne Boyer, LORIA - Nancy Université / KIWI Research team, France
Stainam Brandao, COPPE/Federal University of Rio de Janeiro, Brazil
Stefano Bromuri, University of Applied Sciences Western Switzerland, Switzerland
Vít Bršlica, University of Defence - Brno, Czech Republic
Dumitru Burdescu, University of Craiova, Romania
Diletta Romana Cacciagrano, University of Camerino, Italy
Kenneth P. Camilleri, University of Malta - Msida, Malta
Paolo Campegiani, University of Rome Tor Vergata, Italy
Marcelino Campos Oliveira Silva, Chemtech - A Siemens Business / Federal University of Rio de Janeiro, Brazil
Ozgu Can, Ege University, Turkey
José Manuel Cantera Fonseca, Telefónica Investigación y Desarrollo (R&D), Spain
Juan-Vicente Capella-Hernández, Universitat Politècnica de València, Spain
Miriam A. M. Capretz, The University of Western Ontario, Canada
Massimiliano Caramia, University of Rome "Tor Vergata", Italy
Davide Carboni, CRS4 Research Center - Sardinia, Italy
Luis Carriço, University of Lisbon, Portugal
Rafael Casado Gonzalez, Universidad de Castilla - La Mancha, Spain
Michelangelo Ceci, University of Bari, Italy
Fernando Cerdan, Polytechnic University of Cartagena, Spain
Alexandra Suzana Cernian, University "Politehnica" of Bucharest, Romania
Sukalpa Chanda, Gjøvik University College, Norway
David Chen, University Bordeaux 1, France
Dickson Chiu, Dickson Computer Systems, Hong Kong
Sunil Choenni, Research & Documentation Centre, Ministry of Security and Justice / Rotterdam University of Applied Sciences, The Netherlands
Ryszard S. Choras, University of Technology & Life Sciences, Poland
Smitashree Choudhury, Knowledge Media Institute, The UK Open University, UK
William Cheng-Chung Chu, Tunghai University, Taiwan
Christophe Claramunt, Naval Academy Research Institute, France
Cesar A. Collazos, Universidad del Cauca, Colombia
Phan Cong-Vinh, NTT University, Vietnam
Christophe Cruz, University of Bourgogne, France
Beata Czarnacka-Chrobot, Warsaw School of Economics, Department of Business Informatics, Poland
Claudia d'Amato, University of Bari, Italy
Mirela Danubianu, "Stefan cel Mare" University of Suceava, Romania
Antonio De Nicola, ENEA, Italy
Claudio de Castro Monteiro, Federal Institute of Education, Science and Technology of Tocantins, Brazil
Noel De Palma, Joseph Fourier University, France
Zhi-Hong Deng, Peking University, China
Stojan Denic, Toshiba Research Europe Limited, UK
Vivek S. Deshpande, MIT College of Engineering - Pune, India
Sotirios Ch. Diamantas, Pusan National University, South Korea
Leandro Dias da Silva, Universidade Federal de Alagoas, Brazil
Jerome Dinet, Univeristé Paul Verlaine - Metz, France
Jianguo Ding, University of Luxembourg, Luxembourg
Yulin Ding, Defence Science & Technology Organisation Edinburgh, Australia
Mihaela Dinsoreanu, Technical University of Cluj-Napoca, Romania
Ioanna Dionysiou, University of Nicosia, Cyprus
Roland Dodd, CQUniversity, Australia
Suzana Dragicevic, Simon Fraser University- Burnaby, Canada

Mauro Dragone, University College Dublin (UCD), Ireland
Marek J. Druzdzel, University of Pittsburgh, USA
Carlos Duarte, University of Lisbon, Portugal
Raimund K. Ege, Northern Illinois University, USA
Jorge Ejarque, Barcelona Supercomputing Center, Spain
Larbi Esmahi, Athabasca University, Canada
Simon G. Fabri, University of Malta, Malta
Umar Farooq, Amazon.com, USA
Mehdi Farshbaf-Sahih-Sorkhabi, Azad University - Tehran / Fanavaran co., Tehran, Iran
Anna Fensel, Semantic Technology Institute (STI) Innsbruck and FTW Forschungszentrum Telekommunikation Wien, Austria
Stenio Fernandes, Federal University of Pernambuco (CIn/UFPE), Brazil
Oscar Ferrandez Escamez, University of Utah, USA
Agata Filipowska, Poznan University of Economics, Poland
Ziny Flikop, Scientist, USA
Adina Magda Florea, University "Politehnica" of Bucharest, Romania
Francesco Fontanella, University of Cassino and Southern Lazio, Italy
Panagiotis Fotaris, University of Macedonia, Greece
Enrico Francesconi, ITTIG - CNR / Institute of Legal Information Theory and Techniques / Italian National Research Council, Italy
Rita Francese, Università di Salerno - Fisciano, Italy
Bernhard Freudenthaler, Software Competence Center Hagenberg GmbH, Austria
Sören Frey, Daimler TSS GmbH, Germany
Steffen Fries, Siemens AG, Corporate Technology - Munich, Germany
Somchart Fugkeaw, Thai Digital ID Co., Ltd., Thailand
Naoki Fukuta, Shizuoka University, Japan
Mathias Funk, Eindhoven University of Technology, The Netherlands
Adam M. Gadomski, Università degli Studi di Roma La Sapienza, Italy
Alex Galis, University College London (UCL), UK
Crescenzo Gallo, Department of Clinical and Experimental Medicine - University of Foggia, Italy
Matjaz Gams, Jozef Stefan Institute-Ljubljana, Slovenia
Raúl García Castro, Universidad Politécnica de Madrid, Spain
Fabio Gasparetti, Roma Tre University - Artificial Intelligence Lab, Italy
Joseph A. Giampapa, Carnegie Mellon University, USA
George Giannakopoulos, NCSR Demokritos, Greece
David Gil, University of Alicante, Spain
Harald Gjermundrod, University of Nicosia, Cyprus
Angelantonio Gnazzo, Telecom Italia - Torino, Italy
Luis Gomes, Universidade Nova Lisboa, Portugal
Nan-Wei Gong, MIT Media Laboratory, USA
Francisco Alejandro Gonzale-Horta, National Institute for Astrophysics, Optics, and Electronics (INAOE), Mexico
Sotirios K. Goudos, Aristotle University of Thessaloniki, Greece
Victor Govindaswamy, Concordia University - Chicago, USA
Gregor Grambow, AristaFlow GmbH, Germany
Fabio Grandi, University of Bologna, Italy
Andrina Granić, University of Split, Croatia
Carmine Gravino, Università degli Studi di Salerno, Italy
Michael Grottko, University of Erlangen-Nuremberg, Germany
Maik Günther, Stadtwerke München GmbH, Germany
Francesco Guerra, University of Modena and Reggio Emilia, Italy
Alessio Gugliotta, Innova SPA, Italy
Richard Gunstone, Bournemouth University, UK
Fikret Gurgen, Bogazici University, Turkey

Maki Habib, The American University in Cairo, Egypt
Till Halbach, Norwegian Computing Center, Norway
Jameleddine Hassine, King Fahd University of Petroleum & Mineral (KFUPM), Saudi Arabia
Ourania Hatzi, Harokopio University of Athens, Greece
Yulan He, Aston University, UK
Kari Heikkinen, Lappeenranta University of Technology, Finland
Cory Henson, Wright State University / Kno.e.sis Center, USA
Arthur Herzog, Technische Universität Darmstadt, Germany
Rattikorn Hewett, Whitacre College of Engineering, Texas Tech University, USA
Celso Massaki Hirata, Instituto Tecnológico de Aeronáutica - São José dos Campos, Brazil
Jochen Hirth, University of Kaiserslautern, Germany
Bernhard Hollunder, Hochschule Furtwangen University, Germany
Thomas Holz, University College Dublin, Ireland
Władysław Homenda, Warsaw University of Technology, Poland
Carolina Howard Felicissimo, Schlumberger Brazil Research and Geoengineering Center, Brazil
Weidong (Tony) Huang, CSIRO ICT Centre, Australia
Xiaodi Huang, Charles Sturt University - Albury, Australia
Eduardo Huedo, Universidad Complutense de Madrid, Spain
Marc-Philippe Huget, University of Savoie, France
Chi Hung, Tsinghua University, China
Chih-Cheng Hung, Southern Polytechnic State University - Marietta, USA
Edward Hung, Hong Kong Polytechnic University, Hong Kong
Muhammad Iftikhar, Universiti Malaysia Sabah (UMS), Malaysia
Prateek Jain, Ohio Center of Excellence in Knowledge-enabled Computing, Kno.e.sis, USA
Wassim Jaziri, Miracl Laboratory, ISIM Sfax, Tunisia
Hoyoung Jeung, SAP Research Brisbane, Australia
Yiming Ji, University of South Carolina Beaufort, USA
Jinlei Jiang, Department of Computer Science and Technology, Tsinghua University, China
Weirong Jiang, Juniper Networks Inc., USA
Hanmin Jung, Korea Institute of Science & Technology Information, Korea
Hermann Kaindl, Vienna University of Technology, Austria
Ahmed Kamel, Concordia College, Moorhead, Minnesota, USA
Rajkumar Kannan, Bishop Heber College(Autonomous), India
Fazal Wahab Karam, Norwegian University of Science and Technology (NTNU), Norway
Dimitrios A. Karras, Chalkis Institute of Technology, Hellas
Koji Kashihara, The University of Tokushima, Japan
Nittaya Kerdprasop, Suranaree University of Technology, Thailand
Katia Kermanidis, Ionian University, Greece
Serge Kernbach, University of Stuttgart, Germany
Nhien An Le Khac, University College Dublin, Ireland
Reinhard Klemm, Avaya Labs Research, USA
Ah-Lian Kor, Leeds Metropolitan University, UK
Arne Koschel, Applied University of Sciences and Arts, Hannover, Germany
George Kousiouris, NTUA, Greece
Philipp Kremer, German Aerospace Center (DLR), Germany
Dalia Kriksciuniene, Vilnius University, Lithuania
Markus Kunde, German Aerospace Center, Germany
Dharmender Singh Kushwaha, Motilal Nehru National Institute of Technology, India
Andrew Kusiak, The University of Iowa, USA
Dimosthenis Kyriazis, National Technical University of Athens, Greece
Vitaveska Lanfranchi, Research Fellow, OAK Group, University of Sheffield, UK
Mikel Larrea, University of the Basque Country UPV/EHU, Spain
Philippe Le Parc, University of Brest, France

Gyu Myoung Lee, Liverpool John Moores University, UK
Kyu-Chul Lee, Chungnam National University, South Korea
Tracey Kah Mein Lee, Singapore Polytechnic, Republic of Singapore
Daniel Lemire, LICEF Research Center, Canada
Haim Levkowitz, University of Massachusetts Lowell, USA
Kuan-Ching Li, Providence University, Taiwan
Tsai-Yen Li, National Chengchi University, Taiwan
Yangmin Li, University of Macau, Macao SAR
Jian Liang, Nimbus Centre, Cork Institute of Technology, Ireland
Haibin Liu, China Aerospace Science and Technology Corporation, China
Lu Liu, University of Derby, UK
Qing Liu, The Commonwealth Scientific and Industrial Research Organisation (CSIRO), Australia
Shih-Hsi "Alex" Liu, California State University - Fresno, USA
Xiaoqing (Frank) Liu, Missouri University of Science and Technology, USA
David Lizcano, Universidad a Distancia de Madrid, Spain
Henrique Lopes Cardoso, LIACC / Faculty of Engineering, University of Porto, Portugal
Sandra Lovrencic, University of Zagreb, Croatia
Jun Luo, Shenzhen Institutes of Advanced Technology, Chinese Academy of Sciences, China
Prabhat K. Mahanti, University of New Brunswick, Canada
Jacek Mandziuk, Warsaw University of Technology, Poland
Herwig Mannaert, University of Antwerp, Belgium
Yannis Manolopoulos, Aristotle University of Thessaloniki, Greece
Antonio Maria Rinaldi, Università di Napoli Federico II, Italy
Ali Masoudi-Nejad, University of Tehran, Iran
Constandinos Mavromoustakis, University of Nicosia, Cyprus
Zulfiqar Ali Memon, Sukkur Institute of Business Administration, Pakistan
Andreas Merentitis, AGT Group (R&D) GmbH, Germany
Jose Merseguer, Universidad de Zaragoza, Spain
Frederic Migeon, IRIT/Toulouse University, France
Harald Milchrahm, Technical University Graz, Institute for Software Technology, Austria
Les Miller, Iowa State University, USA
Marius Minea, University POLITEHNICA of Bucharest, Romania
Yasser F. O. Mohammad, Assiut University, Egypt
Shahab Mokarizadeh, Royal Institute of Technology (KTH) - Stockholm, Sweden
Martin Molhanec, Czech Technical University in Prague, Czech Republic
Charalampos Moschopoulos, KU Leuven, Belgium
Mary Luz Mouronte López, Ericsson S.A., Spain
Henning Müller, University of Applied Sciences Western Switzerland - Sierre (HES SO), Switzerland
Susana Munoz Hernández, Universidad Politécnica de Madrid, Spain
Bela Mutschler, Hochschule Ravensburg-Weingarten, Germany
Deok Hee Nam, Wilberforce University, USA
Fazel Naghdy, University of Wollongong, Australia
Joan Navarro, Research Group in Distributed Systems (La Salle - Ramon Llull University), Spain
Rui Neves Madeira, Instituto Politécnico de Setúbal / Universidade Nova de Lisboa, Portugal
Andrzej Niesler, Institute of Business Informatics, Wrocław University of Economics, Poland
Kouzou Ohara, Aoyama Gakuin University, Japan
Jonice Oliveira, Universidade Federal do Rio de Janeiro, Brazil
Ian Oliver, Nokia Location & Commerce, Finland / University of Brighton, UK
Michael Adeyeye Oluwasegun, University of Cape Town, South Africa
Sascha Opletal, University of Stuttgart, Germany
Fakri Othman, Cardiff Metropolitan University, UK
Enn Õunapuu, Tallinn University of Technology, Estonia
Jeffrey Junfeng Pan, Facebook Inc., USA

Hervé Panetto, University of Lorraine, France
Malgorzata Pankowska, University of Economics, Poland
Harris Papadopoulos, Frederick University, Cyprus
Laura Papaleo, ICT Department - Province of Genoa & University of Genoa, Italy
Agis Papantoniou, National Technical University of Athens, Greece
Thanasis G. Papaioannou, École Polytechnique Fédérale de Lausanne (EPFL), Switzerland
Andreas Papasalouros, University of the Aegean, Greece
Eric Paquet, National Research Council / University of Ottawa, Canada
Kunal Patel, Ingenuity Systems, USA
Carlos Pedrinaci, Knowledge Media Institute, The Open University, UK
Yoseba Penya, University of Deusto - DeustoTech (Basque Country), Spain
Cathryn Peoples, Ulster University, UK
Asier Perillos, University of Deusto, Spain
Christian Percebois, Université Paul Sabatier - IRIT, France
Andrea Perego, European Commission, Joint Research Centre, Italy
Mark Perry, University of Western Ontario/Faculty of Law/ Faculty of Science - London, Canada
Willy Picard, Poznań University of Economics, Poland
Agostino Poggi, Università degli Studi di Parma, Italy
R. Ponnusamy, Madha Engineering College-Anna University, India
Jerzy Prekurat, Canadian Bank Note Co. Ltd., Canada
Didier Puzenat, Université des Antilles et de la Guyane, France
Sita Ramakrishnan, Monash University, Australia
Elmano Ramalho Cavalcanti, Federal University of Campina Grande, Brazil
Juwel Rana, Luleå University of Technology, Sweden
Martin Randles, School of Computing and Mathematical Sciences, Liverpool John Moores University, UK
Christoph Rasche, University of Paderborn, Germany
Ann Reddipogu, ManyWorlds UK Ltd, UK
Ramana Reddy, West Virginia University, USA
René Reiners, Fraunhofer FIT - Sankt Augustin, Germany
Paolo Remagnino, Kingston University - Surrey, UK
Sebastian Rieger, University of Applied Sciences Fulda, Germany
Andreas Riener, Johannes Kepler University Linz, Austria
Ivan Rodero, NSF Center for Autonomic Computing, Rutgers University - Piscataway, USA
Alejandro Rodríguez González, University Carlos III of Madrid, Spain
Paolo Romano, INESC-ID Lisbon, Portugal
Agostinho Rosa, Instituto de Sistemas e Robótica, Portugal
José Rouillard, University of Lille, France
Paweł Różycki, University of Information Technology and Management (UITM) in Rzeszów, Poland
Igor Ruiz-Agundez, DeustoTech, University of Deusto, Spain
Michele Ruta, Politecnico di Bari, Italy
Melike Sah, Trinity College Dublin, Ireland
Francesc Saigi Rubió, Universitat Oberta de Catalunya, Spain
Abdel-Badeeh M. Salem, Ain Shams University, Egypt
Yacine Sam, Université François-Rabelais Tours, France
Ismael Sanz, Universitat Jaume I, Spain
Ricardo Sanz, Universidad Politecnica de Madrid, Spain
Marcello Sarini, Università degli Studi Milano-Bicocca - Milano, Italy
Munehiko Sasajima, I.S.I.R., Osaka University, Japan
Minoru Sasaki, Ibaraki University, Japan
Hiroyuki Sato, University of Tokyo, Japan
Jürgen Sauer, Universität Oldenburg, Germany
Patrick Sayd, CEA List, France
Dominique Scapin, INRIA - Le Chesnay, France

Kenneth Scerri, University of Malta, Malta
Rainer Schmidt, Austrian Institute of Technology, Austria
Bruno Schulze, National Laboratory for Scientific Computing - LNCC, Brazil
Ingo Schwab, University of Applied Sciences Karlsruhe, Germany
Wieland Schwinger, Johannes Kepler University Linz, Austria
Hans-Werner Sehring, Tallence AG, Germany
Paulo Jorge Sequeira Gonçalves, Polytechnic Institute of Castelo Branco, Portugal
Kewei Sha, Oklahoma City University, USA
Roman Y. Shtykh, Rakuten, Inc., Japan
Robin JS Sloan, University of Abertay Dundee, UK
Vasco N. G. J. Soares, Instituto de Telecomunicações / University of Beira Interior / Polytechnic Institute of Castelo Branco, Portugal
Don Sofge, Naval Research Laboratory, USA
Christoph Sondermann-Woelke, Universitaet Paderborn, Germany
George Spanoudakis, City University London, UK
Vladimir Stantchev, SRH University Berlin, Germany
Cristian Stanciu, University Politehnica of Bucharest, Romania
Claudius Stern, University of Paderborn, Germany
Mari Carmen Suárez-Figueroa, Universidad Politécnica de Madrid (UPM), Spain
Kåre Synnes, Luleå University of Technology, Sweden
Ryszard Tadeusiewicz, AGH University of Science and Technology, Poland
Yehia Taher, ERISS - Tilburg University, The Netherlands
Yutaka Takahashi, Senshu University, Japan
Dan Tamir, Texas State University, USA
Jinhui Tang, Nanjing University of Science and Technology, P.R. China
Yi Tang, Chinese Academy of Sciences, China
John Terzakis, Intel, USA
Sotirios Terzis, University of Strathclyde, UK
Vagan Terziyan, University of Jyväskylä, Finland
Lucio Tommaso De Paolis, Department of Innovation Engineering - University of Salento, Italy
Davide Tosi, Università degli Studi dell'Insubria, Italy
Raquel Trillo Lado, University of Zaragoza, Spain
Tuan Anh Trinh, Budapest University of Technology and Economics, Hungary
Simon Tsang, Applied Communication Sciences, USA
Theodore Tsiligiridis, Agricultural University of Athens, Greece
Antonios Tsourdos, Cranfield University, UK
José Valente de Oliveira, University of Algarve, Portugal
Eugen Volk, University of Stuttgart, Germany
Mihaela Vranić, University of Zagreb, Croatia
Chieh-Yih Wan, Intel Labs, Intel Corporation, USA
Jue Wang, Washington University in St. Louis, USA
Shenghui Wang, OCLC Leiden, The Netherlands
Zhonglei Wang, Karlsruhe Institute of Technology (KIT), Germany
Laurent Wendling, University Descartes (Paris 5), France
Maarten Weyn, University of Antwerp, Belgium
Nancy Wiegand, University of Wisconsin-Madison, USA
Alexander Wijesinha, Towson University, USA
Eric B. Wolf, US Geological Survey, Center for Excellence in GIScience, USA
Ouri Wolfson, University of Illinois at Chicago, USA
Yingcai Xiao, The University of Akron, USA
Reuven Yagel, The Jerusalem College of Engineering, Israel
Fan Yang, Nuance Communications, Inc., USA
Zhenzhen Ye, Systems & Technology Group, IBM, US A

Jong P. Yoon, MATH/CIS Dept, Mercy College, USA

Shigang Yue, School of Computer Science, University of Lincoln, UK

Claudia Zapata, Pontificia Universidad Católica del Perú, Peru

Marek Zaremba, University of Quebec, Canada

Filip Zavoral, Charles University Prague, Czech Republic

Yuting Zhao, University of Aberdeen, UK

Hai-Tao Zheng, Graduate School at Shenzhen, Tsinghua University, China

Zibin (Ben) Zheng, Shenzhen Research Institute, The Chinese University of Hong Kong, Hong Kong

Bin Zhou, University of Maryland, Baltimore County, USA

Alfred Zimmermann, Reutlingen University - Faculty of Informatics, Germany

Wolf Zimmermann, Martin-Luther-University Halle-Wittenberg, Germany

CONTENTS

pages: 1 - 11

Cross-border and Cross-domain Integration of 3D Content in a European Geospatially Enabled Ecosystem

Lassi Lehto, Finnish Geospatial Research Institute, National Land Survey of Finland, Finland
Jaakko Kähkönen, Finnish Geospatial Research Institute, National Land Survey of Finland, Finland
Jari Reini, National Land Survey of Finland, Finland
Timo Aarnio, National Land Survey of Finland, Finland
Roope Tervo, Finnish Meteorological Institute, Finland

pages: 12 - 23

Decentralised Autonomic Self-Adaptation in a Foraging Robot Swarm

Liam McGuigan, Ulster University, United Kingdom
Roy Sterritt, Ulster University, United Kingdom
George Wilkie, Ulster University, United Kingdom
Glenn Hawe, Ulster University, United Kingdom

pages: 24 - 34

Interpretation Support by Extracting Time Series Classification Patterns using HMM from Text-based Deep Learning

Masayuki Ando, R-GIRO, Ritsumeikan University, Japan
Yoshinobu Kawahara, Institute of Mathematics for Industry, Kyushu University; RIKEN Center for Advanced Intelligence Project, Japan
Wataru Sunayama, School of Engineering, The University of Shiga Prefecture, Japan
Yuji Hatanaka, Faculty of Science and Technology, Oita University, Japan

pages: 35 - 48

Utilizing Fuzzy Sets and Rule Engines for Intelligent Task Assignment in Industry 4.0 Production Processes

Gregor Grambow, Aalen University, Germany
Daniel Hieber, Aalen University, Germany
Roy Oberhauser, Aalen University, Germany

pages: 49 - 59

ARPF - an Augmented Reality Process Framework for Context-Aware Process Execution in Industry 4.0 Processes

Gregor Grambow, Aalen University, Germany
Daniel Hieber, Aalen University, Germany
Roy Oberhauser, Aalen University, Germany
Camil Pogolski, Aalen University, Germany

pages: 60 - 70

A Reference Ontology for Collision Avoidance Systems and Accountability Integrated with DAIDALUS

David Martín-Lammerding, Public University of Navarre (UPNA), Spain
José Javier Astrain, Public University of Navarre (UPNA), Spain
Alberto Córdoba, Public University of Navarre (UPNA), Spain

pages: 71 - 82

Automatic Recognition of Continuous Sign Language for Public Services

Robson Silva de Souza, University of Campinas, Brazil
José Mario De Martino, University of Campinas, Brazil
Janice Gonçalves Temoteo Marques, University of Campinas, Brazil
Ivani Rodrigues Silva, University of Campinas, Brazil

Cross-border and Cross-domain Integration of 3D Content in a European Geospatially Enabled Ecosystem

Lassi Lehto, Jaakko Kähkönen
 Department of Geoinformatics and Cartography
 Finnish Geospatial Research Institute
 Masala, Finland
 e-mail: lassi.lehto@nls.fi, jaakko.kahkonen@nls.fi

Jari Reini^a, Timo Aarnio^a, Roope Tervo^b
 National Land Survey of Finland^a
 Finnish Meteorological Institute^b
 Helsinki, Finland
 e-mail: jari.reini@nls.fi, timo.aarnio@nls.fi,
 roope.tervo@fmi.fi

Abstract—The Geospatially Enabled Ecosystem for Europe (GeoE3) project has as the main goal to develop use case-oriented, cross-border and cross-domain geospatial services conforming to the second-generation interface standards. The five-nation project focusses on cross-domain integration of content by applying dynamic, service level methods for joining climate and statistical data with geospatial features. Important climate parameters have been successfully integrated with building data to support the envisaged use cases. Various technologies are being tested for providing seamless, cross-border access to geospatial resources in context of the new Open Geospatial Consortium’s OGC API family of service interface standards. 3D aspects of geospatial data are also being considered, both from content encoding and service interface point of view. Both pre-created 3D data sources and dynamic, on-the-fly techniques are applied in the 3D content provision. The project aims at supporting renewable energy- and smart city-related applications.

Keywords—*geospatial ecosystem; cross-border; cross-domain; climate; OGC API; 3D; solar energy; smart city.*

I. INTRODUCTION

In an interoperable co-operative computing environment, every participating actor should gain benefits. This is the main guiding principle in the spatial data ecosystems currently being developed. A major action, Geospatially Enabled Ecosystem for Europe (GeoE3), is working to establish an ecosystem of use case-oriented geospatial services in five European countries [1]. The action is part of the Connecting Europe Facility (CEF) programme advancing European transport, energy and digital infrastructure development [2]. The project runs for three years and involves five national mapping and cadastral agencies, together with a few other governmental organizations and private companies [3]. The main goals of the project include adaptation of modern geospatial APIs (Application Programming Interfaces) to establish cross-border services for renewable energy and urbanization related use cases, use of dynamic service-level mechanisms for cross-domain content integration, and promotion of the developed approaches through extensive online innovation and education programme.

GeoE3 consortium membership includes the national mapping or cadastral agency from Finland, Norway, Estonia,

The Netherlands and Spain. Furthermore, the consortium includes the national meteorological and statistical agency of Finland: Finnish Meteorological Institute and Statistics Finland, respectively. Private companies include Spatineo from Finland and Aventi Intelligent Communication from Norway. Open Geospatial Consortium (OGC) Europe participates in the project as a representative of the standardization community. National Land Survey of Finland coordinates the project that will be finalized by Sep 2023 [4].

The main use cases identified by GeoE3 include renewable energy applications, specifically in the context of buildings construction and use. In particular, this involves solar energy potential, based on solar panels on rooftops. Wind energy and heating/cooling facilities of the buildings are considered too. The other use cases deal with traffic applications, in particular efficient use of electric cars, and with various urbanization challenges. The last one is considering urban expansion efficiency employing the United Nation’s Sustainable Development Goal (SDG) indicator 11.3.1 ‘Ratio of land consumption rate to population growth rate’ [5].

The paper is organized as follows. In Section II, the envisioned service architecture of the GeoE3 platform is described. Section III discusses various mechanisms for content integration. In Section IV, some novel ideas concerning OGC API Features functionality are discussed. Section V describes results of the coverage data integration experiments. Section VI presents the initial considerations of the project on dealing with 3D geodata and Section VII describes the first results achieved. The paper ends with conclusions in Section VIII.

II. SERVICE ARCHITECTURE

The GeoE3 project aims at establishing a set of use case-oriented services that enable content integration both across national borders and domain boundaries. The idea is to set up an integration layer on top of national services. The integration layer provides modern service interfaces to the client side and accommodates various categories of source services on the country level. The GeoE3 services are based on the OGC API family of second-generation, internationally standardized service interfaces [6]. These include services like OGC API Features [7], OGC API Coverages [8] and OGC API Records [9]. Internally the GeoE3 platform will apply at least OGC

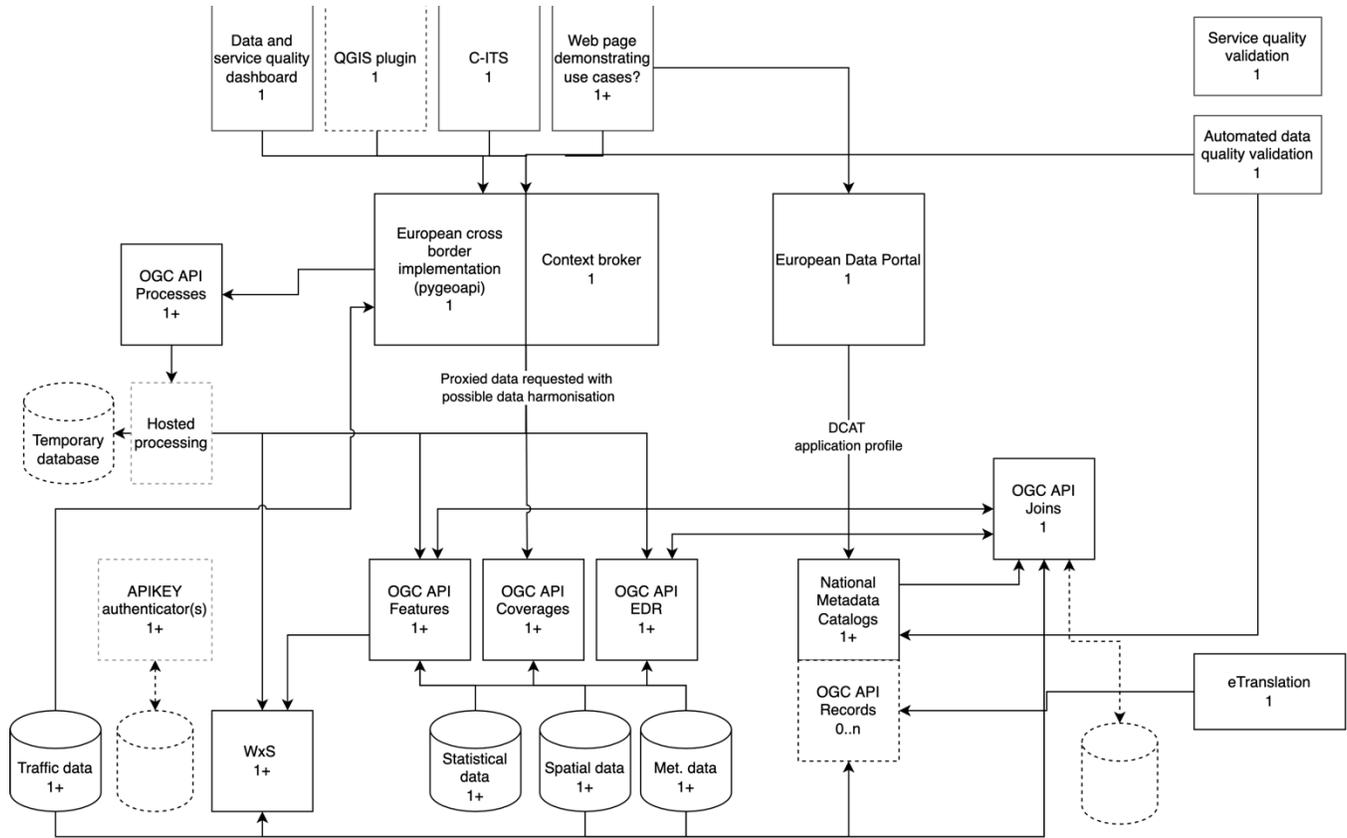


Figure 1. The GeoE3 service architecture.

API Processes [10] for introducing analysis-based content layers to the clients and OGC API Joins [11] for content integration across domain boundaries. OGC API Environment Data Retrieval (EDR) [12] interface will be used for requesting meteorological data, particularly for supporting transport-related applications.

The illustration in Fig. 1 shows the GeoE3 service architecture. The WxS box in the left bottom corner represents all the country level data resources that will mostly be accessed through service access points conforming to the legacy OGC standard interfaces, like Web Feature Service (WFS), Web Coverage Service (WCS), and the OGC API family of services. The databases at the bottom emphasize the GeoE3 goal of integrating spatial data with statistical and other tabular data sets. All the data resources are made available via modern interface standards and are integrated in cross-border manner on the GeoE3 integration platform (European cross-border implementation in Fig. 1).

Metadata records of all GeoE3 services will be harvested from the national metadata catalogues into the European Data Portal (EDP) and also be available from the OGC API Records service interface maintained on the GeoE3 platform. The CEF eTranslation [13] building block tool will be used to provide translations between the national language and the target language. It provides machine-to-machine use via a Web service interface for translating text and documents in the

multilingual European context. The eTranslation Web service is asynchronous. In other words, the client sends a translation request and is notified later, when the text snippet or document is translated. The client needs to expose a call-back URL, which will receive a notification that the translation job in question has been completed. Translated metadata documents will be published via OGC API - Records interface.

An implementation of the OGC API Joins standard is an essential component of the GeoE3 service architecture. It will be responsible for joining tabular data sets with geospatial features. The resulting combined resource is then made accessible to client applications via an OGC API Features service.

The GeoE3 integration platform is a cloud service-based solution for aggregated access to all content becoming available in the context of the project. Django is used as the Web application framework on the platform [14]. The main geospatial software package used on the platform for the provision of OGC API compliant services is a product called pygeoapi [15]. pygeoapi is a Python library supporting OGC API Features and partially supporting OGC API Coverages, OGC API Records and OGC API Processes interfaces. pygeoapi is the reference implementation for the OGC API Features specification.

In the GeoE3 project, pygeoapi has been tailored in many ways. First of all, it has been adapted to be used in the Django framework. Map visualisation library originally used in the

product, Leaflet, has been replaced by more feature-rich OpenLayers [16]. Fundamental changes have been made to pygeoapi's handling of OGC API Features HTML-formatted output (see Section IV). OGC API Coverages implementation has been enhanced to support Web-based viewing of the coverage data. This involves for example support for PNG images and the 'scale-size' parameter in the coverage request. A set of content provider plugins for various different data sources have also been developed. Some work has been invested in developing more advanced functionalities for the Web user interface, for example to support 3D viewing of content (see Section VII).

III. CONTENT INTEGRATION

A. Content Integration with OGC API Features

An essential component on the GeoE3 integration layer is an OGC API Features service implementation. The successor for the OGC's WFS interface specification, OGC API Features follows the principles of the REST (Representational State Transfer) service architecture. The output of the service consists of individual geospatial features, organized as a GeoJSON -encoded feature collection [17], geometries expressed in WGS84 Coordinate Reference System (CRS). The OGC API Features service instance offers a set of data collections, accessible from the given service address paths. This provides a natural setting for organizing individual countries' data sets as collections inside a single OGC API Features instance. For instance, the buildings data sets of Finland, Norway and Spain could be organized as follows.

- /geoe3/buildings/collections/buildings_FI/items
- /geoe3/buildings/collections/buildings_NO/items
- /geoe3/buildings/collections/buildings_ES/items

The service implementation adopted in the GeoE3 platform, pygeoapi, provides a flexible plugin architecture for content provision and formatting. The off-the-shelf library supports, via OGR Simple Features Library [18] and rasterio [19] provider plugins, a vast array of vector and raster source formats and services. For example, WFS is supported as source service. OGR provider also supports OGC API Features service access, thus enabling cascading service approaches. Specific project-related needs can be accommodated by developing a tailored plugin. As the plugin architecture is based on Python language, all the Python-based geospatial libraries can be easily utilized. The set of already available preliminary GeoE3 components include plugins for WFS versions 1.1.0 and 2.0.0 with basic schema transformation capabilities, a plugin for WCS version 1.0.0, a plugin for a remote OGC API Features service and a plugin for accessing raster data from a virtual file (VRT) -based data storage.

Schema transformations form an essential part of the functionality of a vector data provider plugin. The principles of the so-called alternative encoding INSPIRE (Infrastructure for Spatial Information in Europe) schema [20] is used as the approach for the common target schema in GeoE3. So far the focus has mostly been on the content theme 'Buildings'. Schema transformations have been developed from two national schemas (Finland and The Netherlands), from two

legacy INSPIRE schemas (Norway and Spain), and from one INSPIRE-like schema (Estonia) to the common target schema. The transformation also involves translating the data elements from the GML (Geography Markup Language) encoding to the GeoJSON encoding, in case of four countries. The property names of the current version of the target schema for theme 'Buildings' are presented in Table 1.

TABLE 1. GEOE3 TARGET SCHEMA

inspireId_localId
inspireId_namespace
inspireId_versionId
beginLifespanVersion
endLifespanVersion
externalReference_informationSystem
externalReference_informationSystemName
externalReference_reference
conditionOfConstruction
dateOfConstruction
currentUse_currentUse
currentUse_currentUse_href
currentUse_percentage
address_adminUnit_name_[lang]
address_postCode
address_thoroughfare_name_[lang]
address_locator_designator_addressNumber
cadastralParcel_label
officialArea_value
floorDescription_floorArea
heightAboveGround_value
heightAboveGround_status
heightAboveGround_status_href
volume
numberOfFloorsAboveGround
numberOfBuildingUnits
numberOfDwellings
heatingSystem
heatingSource
materialOfStructure
constructionMethod
materialOfFacade
geometry2D_referenceGeometry
geometry2D_horizontalGeometryReference
geometry2D_horizontalGeometryReference_href
geometry2D_verticalGeometryReference
geometry2D_verticalGeometryReference_href

The INSPIRE-defined model simplification rules applied in the schema definition process include General Flattening, Associated Component Soft Type, Simple Codelist Reference and Simple Geographic Name [21]. Two property names in the schema cannot be derived from INSPIRE data models: ‘volume’ and ‘constructionMethod’. These names can be taken as proposals for a further extension of INSPIRE schemas.

Another important consideration in geospatial data integration is the use of Coordinate Reference Systems. According to the GeoJSON specification, the CRS to be used in GeoJSON-compliant files is WGS84 with axis order: longitude, latitude (identifier: urn:ogc:def:crs:OGC::CRS84). Consequently, all queries to the GeoE3 integration platform’s OGC API Features services must be expressed in this CRS. However, all the source data sets of the OGC API Coverages services are stored in the national CRSs and queries are thus expected in these CRSs. In the GeoE3 project a decision has been made to use Pseudo Mercator (identifier: urn:ogc:def:crs:EPSG::3857) as the common CRS on the client side. OpenLayers, the map application library used in the current user interface, is responsible for carrying out the required CRS transformations to and from the client side CRS.

B. Content Integration with OGC API Processes

OGC API Processes is a new resource-based process definition standard. The standard defines how the client program can start the process, how the process inputs are provided, and where the results of the process are stored.

Computational tasks are wrapped inside the process and the results are provided to the client application. Processes can be for example geometry buffering or routing from point A to point B. The process can be either synchronous, giving the results immediately, or asynchronous, in which case the service informs the client, when the results are ready and where they can be found.

The GeoE3 project has developed a process, in which the 3D geometry of the building is dynamically generated. In addition, a process is being developed to calculate the average amount of solar energy in a building area. This dynamically created information can then be integrated back to the original features.

The OGC API Processes interface standard follows the same principles as the other OGC API standards (e.g., OGC API Features). The path structure of the GeoE3 OGC API Processes services for theme ‘Buildings’ might be as follows.

- /geoe3/buildings (The service landing page)
- /geoe3/buildings/conformance (Conformance classes describing what service can do)
- /geoe3/buildings/processes (The list of processes in the service)
- /geoe3/buildings/processes/extrude-building (Process to create 3D buildings from the building footprint)
- /geoe3/buildings/processes/solarenergy-potential (Solar energy potential calculator)

The first application of OGC API Processes -based functionality in the GeoE3 project is the dynamic creation of LOD1 (LOD: Level of Detail) category 3D models of buildings. The on-the-fly computation has been implemented on the GeoE3 integration platform and is launched automatically in case only part of the buildings is available as pre-created LOD2 models (in case of Finland). If 3D buildings do not exist at all, the on-the-fly computation is applied always (in case of Norway).

The computation uses as input the 2D building footprints, the Digital Terrain Model (DTM) and Digital Surface Model (DSM) elevation values from within the building polygon. When the user selects an individual building in the client application, the corresponding feature data is first requested from the GeoE3 OGC API Features service using the buildings feature identifier. The 2D geometry is extracted from the received feature and used as input for requests to retrieve DTM and DSM values contained within the building outline polygon. The lowest DTM value is used to determine the elevation of the building’s floor. An average of the DSM values is used for elevation of the roof. To minimise the possible deviation caused by nearby trees, only lower half of the DSM values are used. Finally, a CityJSON encoding of the LOD1 building model is generated using the building’s footprint polygon and the computed elevations as input. An example of the resulting 3D view is shown in Fig. 2. As another example, in Fig. 3 dynamically created LOD1 building models are shown with bluish colour, together with buildings in genuine, pre-created LOD2 models with red coloured roofs.

C. Content Integration with OGC API Joins

OGC API Joins provides a method for joining spatial and tabular data. The standard is currently in draft status in OGC and is going to be used in GeoE3. It allows integration of tabular statistical data to the platform.

To carry out the join, the service needs the spatial and tabular data and a common identifier between them. Several columns of the tabular data can be joined at once. Both the spatial and tabular data can either be configured to be available from the server or they can be uploaded on demand. Spatial data can thus be delivered via OGC Features API services and tabular data, e.g., via PxWeb API [22] compliant services.

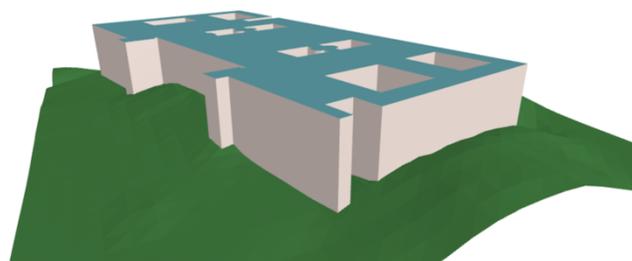


Figure 2. An example of a LOD1 model of a building (the Finnish House of Parliament), created in an on-the-fly OGC API Processes -based computation. Shown together with DTM.



Figure 3. Buildings with on-the-fly computed LOD1 models (bluish roofs), shown together with buildings represented with a pre-created LOD2 model (red coloured roofs) in the 3D viewing component (in the right bottom corner).

The goal of the project is to evaluate the fitness for purpose of the OGC API Joins standard in the use cases identified in the project. An OGC API Joins instance will be set up and configured, along with a browser-based end user interface. While OGC API Joins can be demonstrated and used with a user interface, it can be more valuable when used in the background as a part of an automated process. An obvious benefit of joining data is that spatial data need not be stored in several places.

The output of OGC API Joins has several options. The combined data can be downloaded for further use in several spatial formats, such as GeoJSON. Furthermore, results from the joining operation can be calculated from the data. This includes information about successful joins, unsuccessful joins and extra data rows. In the project this derived data will be visualized in the user interface as a report for the user.

IV. OGC API FEATURES RETHOUGHT

Some of the OGC API Features concepts have been rethought in the development of the preliminary GeoE3 platform. For instance, the HTML-formatted feature browsing in the OGC API Features development is generally understood as a tabular presentation of feature properties, browsed through the feature collection page by page. When there are potentially millions of features in the national databases, and when the order of features presented is arbitrary, the browsing process often becomes unfeasible. The feature browsing in the GeoE3 OGC API Features is returned back to the traditional spatially organized, map-based browsing. A plain background map is first shown to the user. The individual vector features are requested and presented only when the user zooms in deep enough.

Another new concept introduced in the GeoE3 project is the idea of treating the HTML-formatted visualization of an individual spatial feature as an application-specific dashboard of the feature [23]. Into this dashboard, a wide set of feature properties and visualizations can be collected by the GeoE3 integration platform. For example, a sun energy-related building dashboard could contain all the relevant building attributes - like the area, volume and heating information, and the DSM-

based sun exposure analysis from the building's area - useful for solar energy analyses. Furthermore, climate-related information could be integrated to the same dashboard on the GeoE3 platform by accessing appropriate values from a meteorological service. The dashboard could be seen as a shop window for the available content, and could later be downloaded into a digital analysis process in form of the corresponding JSON-formatted data set. An example of a prototypical building dashboard for sun energy-energy related applications is shown in Fig. 4. OGC API Features browsing by map view is shown on the left. On the right, a detail window of the selected feature is shown as 2D and 3D representations. Further on, the full set of feature attributes is displayed, together with a detailed 2D map with orthophoto as the background layer. The third user interface presents the building in the full-size 3D viewing module.

The sun exposure computation used in the GeoE3 feature dashboard is based on the geomorphometric analysis function 'TimeInDaylight' of WhiteBoxTools [24]. The calculation is based on the use of the DSM values around the building. In the server side the process is currently organized as a cascading Web Map Service (WMS). The process providing the source image retrieves the required part of the DSM data

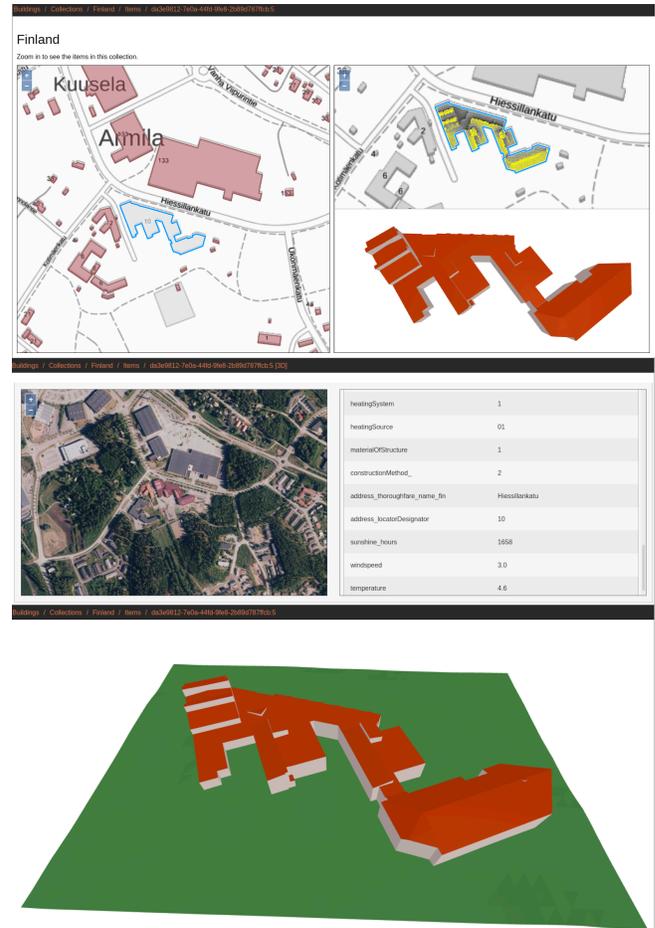


Figure 4. Preliminary OGC API Features html output from the GeoE3 platform as a building's renewable energy dashboard.

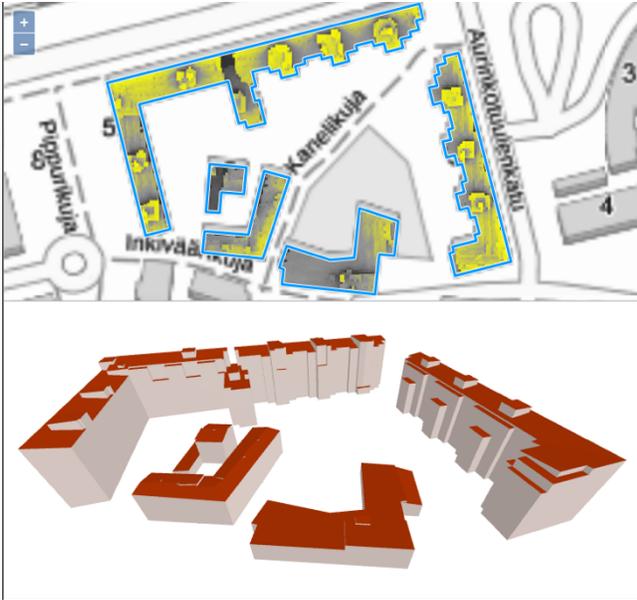


Figure 5. The result of the sunlight exposure analysis of the selected group of buildings (top), together with a LOD2 3D model viewing component (bottom).

from the GeoE3 OGC API Coverages service (see Section V). The resulting exposure information is formatted as TIFF image, which is then converted by the WMS service to a PNG image for viewing in the Web client environment. The display of the analysis results is presented in Fig. 5. In future the analysis is planned to be implemented and further developed as a process inside the OGC API Processes service implementation of the GeoE3 platform (see Section IIIB).

V. COVERAGE DATA

The GeoE3 project is also involving coverage data in its content integration development. This is somewhat novel approach, as most of the previous projects aiming at geodata harmonization and integration have been focusing on vector-formatted data sets. The first coverage data sets being dealt with in the GeoE3 project include themes like elevation and climate.

A. Elevation data

Given the envisaged use case on solar energy potential of a building rooftop, DSM becomes an important source data set. Based on the DSM values, the form of the roof can be automatically derived. The amount of sun exposure can be also evaluated, based on the DSM of the surrounding areas. DTM is important for determining the absolute elevation of the building's floor level, which is needed for instance in volume calculations and integrated visualizations.

The OGC API Coverages service interface is implemented in the GeoE3 integration platform's coverage-related services. The OGC API Coverages is currently available as a draft specification and is thus not yet widely implemented. However, the pygeoapi software package, used in GeoE3 integration platform, provides preliminary support for the specification.

In the project the OGC API Coverages functionality of pygeoapi has been extended in various ways. Provider plugins have been developed to support access to WCS-based source services. Support for PNG output has been added to support visualization in Web clients. An important addition is the possibility to scale the output grid to the resolution appropriate for the client's needs. This is achieved by supporting the 'scale-size' parameter in the coverage query.

As with the OGC API Feature case, explained in Section IV, the Web based access to the OGC API Coverages service interface has been extended in the GeoE3 project. The goal is to facilitate easy viewing of the coverage contents in the Web user interface. To enable this, support for HTML formatted output has been added to pygeoapi. The initial output for the coverage content request in HTML format is an HTML page with an empty map window. The client application can then first perform a request for a background map and then send back a query for a PNG-formatted coverage. The initial view covers the full extent of the coverage. The background map will then enable the user to zoom in and pan the map appropriately to the location the user is interested in. Finally, the user can download the part of the coverage visible in the map view in desired coverage format (GeoTIFF and CoverageJSON [25] being the formats available at the moment).

Data integration across borders is one of the main goals of the GeoE3 project. In case of coverage data, this is particularly interesting, as little work has previously been done in this area. Coverage integration on the GeoE3 integration platform is based on the experimental implementation of cross-collection query as defined in the OGC API Features specification. In this approach the query has a new 'collections' parameter, by which a list a relevant collection names can be given. All the other parameters are applied in all the coverages listed in the 'collections' parameter. Thus, if the bounding box of the query overlaps a national border and the countries on both sides of the query are included into the 'collections' parameter value, the result would be a grid, in which the both coverages are included. As an example, the following query to the GeoE3 DSM Coverages interface will produce the result shown in Fig. 6:

```
/geoe3/dsm/search?f=png&collections=DSM_NO,DSM_FI&subset=x(1225341:3876849),y(9938802:11883241)&scale-size=x(1500),y(1100)
```

Production of the integrated image in Fig. 6 involves quite a lot of processing on the server side. The incoming query's bounding box, expressed in the Pseudo Mercator CRS (EPSG:3857) must first be transformed to the CRS of source data sets. When the requested grid is received from the GeoE3 platform's corresponding OGC API Coverages service, the grid must be reprojected to the target CRS, i.e., Pseudo Mercator. Once both countries' grids have been queried, they will be merged into a single result grid, considering the nodata areas present in the source grids. If the request is for a visible image, the grid needs to be masked for nodata areas and formatted in PNG. Each result image is also stretched to full radiometric scale for the best possible visual exploration.



Figure 6. Response image for a cross-border (Finland-Norway) request for DSM utilising cross-collection query capabilities of the OGC API Coverages interface (experimental).

B. Climate data

Climate-related information was selected as one of the main target areas of the project, based on the requirements of the envisaged use cases. The most interesting climate parameters include sunshine hours, wind conditions and mean temperatures.

It was decided to approach the climate parameters as coverage data, although the source information is typically available as point values of weather observation stations. From the GeoE3 integration platform the climate data is thus available via an OGC API Coverages service endpoint. The first implementation, available currently only for the area of Finland, is based on the idea of cached grid data that is computed from the original observation points.

When a request for climate coverage data comes into the integration platform, the process will first check, whether the cached GeoTIFF file for the requested climate parameter exists on the platform. If the file is not found, a request is made to the OGC API Features service of the Finnish Meteorological Institute (FMI) for the corresponding point data. The data set is requested with a bounding box covering the whole country. From the received value point set, a gridded representation is derived using the SciPy library's [26] 'griddata' function. The process is first run with the method 'nearest' to cover the whole request area. Then the same is carried out with the method 'linear', which only covers the area spanned with the observation points. Subsequently, the two grids are combined, so that the grid points calculated with 'linear' take precedence. Finally, the grid is cropped with the boundary of the country to get the final result, which is then saved on the platform as a GeoTIFF file. The quality of the resulting grid is heavily dependent on the number and layout of the observation stations in the area. Best results have so far been achieved with temperature data.

When further queries come in, for instance as the user zooms in the map view, the cached GeoTIFF file is accessed to compute the result. This significantly speeds up processing of the queries. A functionality has been added to the Web user

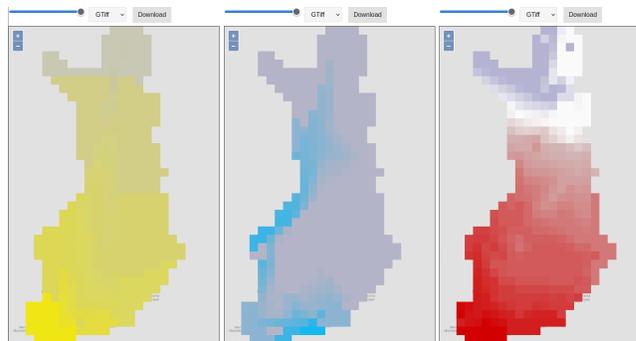


Figure 7. Climate coverages displayed in the GeoE3 Web user interface: sunshine hours (left), windspeed (middle) and mean temperature (right). All parameters are averages over 30 years.

interface for displaying values of the visualized parameter when the mouse is hovered over the map. These queries are carried out as OGC API Coverages requests with zero-size bounding box and are performed on the cached GeoTIFF file on the server.

The climate grid is visualized in the Web user interface with colored raster, requested from the OGC API Coverages service as a PNG image. Examples of the user interface are presented in Fig. 7. The real grid data can be downloaded from the viewed area in GeoTIFF and CoverageJSON formats.

VI. PRELIMINARY 3D WORK

The GeoE3 project introduces the idea of an HTML-formatted application-specific dashboard, especially in the context of building construction and use with renewable energy applications, such as solar energy potential. While computing the energy potential, many data sources and service interfaces can be utilized, depending on the computing tasks in hand. Using DTMs, DSMs and building footprints, one can automatically compute a 2D raster layer that indicates rooftops, suitable for solar energy production. This raster layer can then be portrayed together with the DSM and the orthophoto for 3D visual exploration, even without proper 3D models of the buildings.

In the exemplary visualization shown in Fig. 8, locations suitable for solar panels are presented with yellow colour. The visualization is a result of a parameterized on-the-fly process, in which the DSM slope and aspect values are used, together with the building footprints, and the height difference between the DSM and the DTM, to compute the areas well exposed to sun energy. Fig. 8 shows a three.js-based [27] Web application that requests the analysis result layer and the DSM from OGC-compatible service interfaces and drapes the raster on top of the DSM.

One of the goals of the GeoE3 project is to make use of 3D vector data. For instance, the project coordinator, National Land Survey of Finland, has made first 3D building models available from limited test areas. The production process is based on the use of the building footprints and DSM, and aims at constructing LOD2 category building geometries.



Figure 8. Visual exploration of potential solar panel locations on rooftops. Analysis is based on the use of DSM slope and aspect values, height differences between the DSM and the DTM, and building footprints.

A widely accepted, production-level service interface specification for 3D models does not exist yet. A possible workaround for this situation is to store the 3D geometries into a 3D City DB database [28] and export them as a CityGML [29] or CityJSON [30]-formatted file on demand.

At this moment, CityJSON format is the most interesting proposal for the 3D vector representation. CityJSON is a JSON-based subset of the OGC's CityGML data model and is designed for storage and transfer of 3D city models. CityJSON has been standardized by the OGC as an official OGC Community Standard in Oct 2021. Current version of the CityJSON standard is available at the CityJSON development site.

There are not yet freely available, browser-based applications for visualization of CityGML content. However, JSON processing is widely supported in browsers, and CityJSON can naturally benefit from this too [31]. There are development initiatives for providing an easy-to-use CityJSON-supporting software component for Web applications. On the left side in Fig. 9, an example is presented of browser-based visualization displaying NLS Finland buildings in CityJSON format utilizing a software component called Ninja [32]. On the right side in Fig. 9, the same features are presented using the 3D map view of the QGIS application [33], imported with the help of the QGIS plugin for CityJSON.

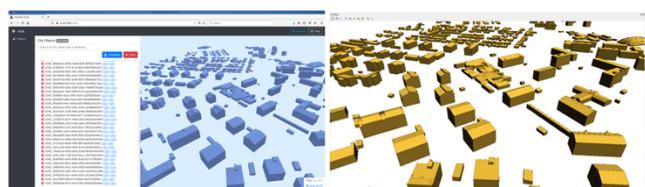


Figure 9. Visualization of 3D vector buildings in browser using Ninja (on the left) and in the 3D map view of the QGIS application (on the right).

VII. 3D EXPERIMENTS

In the GeoE3 project, a substantial amount of work has been devoted on developing functionalities for 3D representation and visualization of buildings. Due to varying level of maturity and techniques applied in participating countries, a

set of different approaches has been used to achieve a reasonably consistent provision of 3D buildings for all the countries.

In the case of Finland, Estonia and The Netherlands, pre-existing LOD2 category 3D models were available as downloadable files. They are encoded in various formats (database dump, CityGML, CityJSON) and were downloaded and imported into a database on the GeoE3 integration platform. In case of Norway, no pre-existing 3D data set could be found and thus an on-the-fly process for producing 3D models in real time was developed (described in Section III B). The case of Spain is different from the others as Spain already has a service endpoint available providing access to 3D representation of the country's buildings encoded in Keyhole Markup Language (KML). This KML-encoded 3D building model is produced by an on-the-fly process using floor plans as the source information. Consequently, the resulting models are in LOD1 category. A proxy service was developed on the GeoE3 integration platform to carry out the required KML to CityJSON transformation. A schematic illustration of the 3D buildings provision framework of the GeoE3 project is shown in Fig. 10.

As a final result of the development, 3D representation of buildings in all the participating countries is available on the GeoE3 platform, consistently encoded in CityJSON, and retrievable using the building's feature identifier as the key. The GeoE3 Web user interface currently utilizes a 3D visualization component 'ThreeJsViewer' developed by the University of Delft [34]. Some representative examples are shown in Fig. 11.

VIII. CONCLUSIONS AND FUTURE WORK

GeoE3 project has as its main goal the development of a geospatially enabled ecosystem of use case-oriented services conforming to modern, second-generation interface standards. The project is tackling the challenge of cross-border and cross-domain integration of content employing a flexible, state-of-the-art integration platform, based on OGC API supporting Python library called pygeoapi. The plugin architecture of pygeoapi facilitates development of tailored components for connecting the platform to various different country level source services.

The overall service architecture of the GeoE3 project can be seen as a three-tier architecture with country level services forming the bottom data layer. The GeoE3 integration platform with advanced capabilities for dynamic content integration and use case-specific adaptation of the resulting data sets works as the middle processing layer. Finally, end user applications form the third architecture layer.

An application-specific output from an OGC API Features service can be provided in the form of an html-formatted dashboard of the feature in focus. Based on the user needs, an appropriate set of information items can be collected and put together by the GeoE3 platform, and the resulting data set be formatted in HTML as a shop window for potential users. The same data set could finally be requested as a JSON-encoded package, to be fed into external analysis processes.

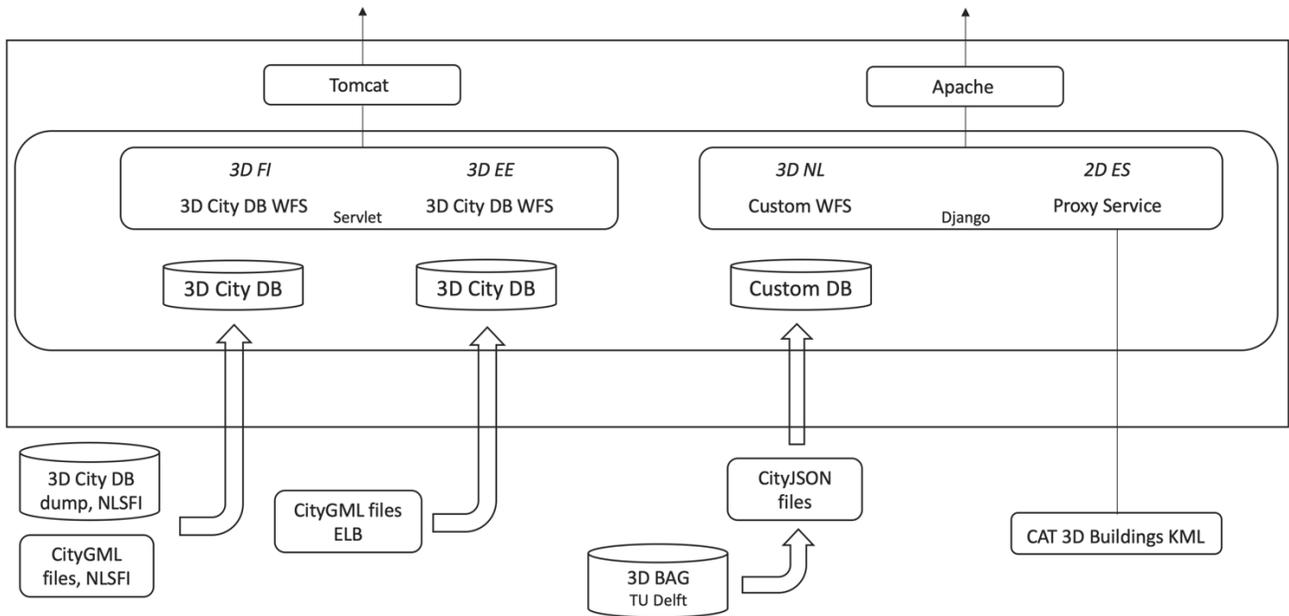


Figure 10. The 3D buildings data provision framework of the GeoE3 project. On-the-fly processes for creating LOD1 models are not included. (FI: Finland, EE: Estonia, NL: The Netherlands, ES: Spain, NLSFI: National Land Survey of Finland, ELB: Estonian Land Board, TU Delft: Delft University of Technology, CAT: General Directorate of Cadastre, Spain).

One of the main results of the GeoE3 work so far is the availability of 3D buildings from all the participating countries. The mechanisms behind the service interface vary a lot from country to country, but as the final end result 3D buildings are accessible by feature ID queries via a WFS conforming service in CityJSON encoding. A real-time process has been developed for generating LOD1 models where LOD2 building models are not available. The process is based on the building's 2D footprint and DTM and DSM coverages within the footprint and provides CityJSON encoded LOD1 models

via an OGC API Processes -compliant service implementation.

First climate-related data sets have also become available on the GeoE3 integration platform. These provide 30-year averages of sunshine hours, wind speed and mean temperature as coverage type content from an OGC API Coverages service.

The project will continue developing the described functionalities over the coming one year and half and aims at wide adoption of the developed services by actors working in the identified use case domains.

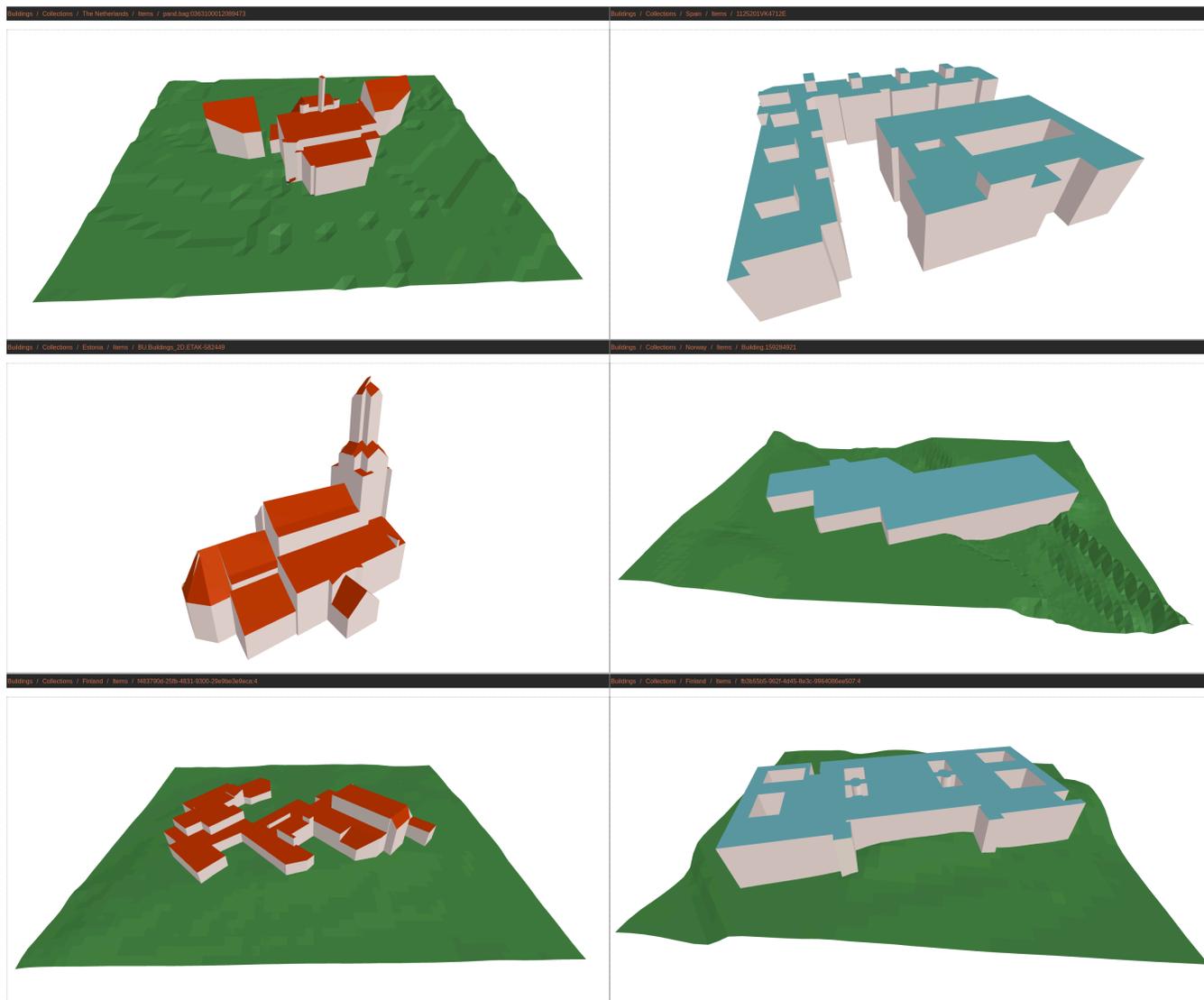


Figure 11. Examples of 3D buildings from GeoE3 participating countries. The Netherlands (top left), Spain (top right), Estonia (middle, left), Norway (middle right) and Finland (bottom). The buildings from Spain, Norway and Finland (bottom right) presented as dynamically created LOD1 models. DTM missing from Spain (building model not in absolute elevation) and Estonia (DTM data not yet available).

ACKNOWLEDGMENT

The GeoE3 project is co-financed by the Connecting Europe Facility (CEF) of the European Union with the grant agreement number INEA/CEF/ICT/A2019/2063390.

The authors wish to acknowledge CSC – IT Center for Science, Finland, for computational resources.

REFERENCES

- [1] L. Lehto, J. Kähkönen, J. Reini, T. Aarnio, and R. Tervo, Cross-border and Cross-domain Integration of Content in a European Geospatially Enabled Ecosystem. *GEOProcessing 2021, the Thirteenth International Conference on Advanced Geographic Information Systems, Applications, and Services*, Jul 18–22, 2021, Nice, France, pp. 1–5. ISBN: 978-1-61208-871-6
- [2] CEF, Connecting Europe Facility, Home Page. [Online]. Available from: <https://ec.europa.eu/inea/en/connecting-europe-facility>, 2021, [retrieved: May, 2022]
- [3] GeoE3, Geospatially Enabled Ecosystem for Europe, GeoE3 Home Page. [Online]. Available from: <https://geo3.eu>, 2021, [retrieved: May, 2022]
- [4] NLS, National Land Survey of Finland Home Page. [Online]. Available from: <https://www.maanmittauslaitos.fi/en>, 2021, [retrieved: May, 2022]
- [5] United Nations, SDG Indicators, Goal 11. [Online]. Available from: <https://unstats.un.org/sdgs/metadata/?Text=&Goal=&Target=11.3>, 2021, [retrieved: May, 2022]
- [6] OGC, OGC API Home Page. [Online]. Available from: <https://ogcapi.ogc.org>, 2021, [retrieved: May, 2022]
- [7] OGC, OGC API – Features Home Page. [Online]. Available from: <https://ogcapi.ogc.org/features/>, 2022, [retrieved: May, 2022]
- [8] OGC, OGC API – Coverages Home Page. [Online]. Available from: <https://ogcapi.ogc.org/coverages/>, 2022, [retrieved: May, 2022]
- [9] OGC, OGC API – Records Home Page. [Online]. Available from: <https://ogcapi.ogc.org/records/>, 2022, [retrieved: May, 2022]
- [10] OGC, OGC API – Processes Home Page. [Online]. Available from: <https://ogcapi.ogc.org/processes/>, 2022, [retrieved: May, 2022]
- [11] OGC, OGC API – Joins, Draft. [Online]. Available from: <https://github.com/opengeospatial/tjs/blob/master/drafts/OGC%20API%20-%20Joins%20-%20Part%201%20Core.pdf>, 2022, [retrieved: May, 2022]
- [12] OGC, OGC API – Environmental Data Retrieval (EDR) Home Page. [Online]. Available from <https://ogcapi.ogc.org/edr/>, 2022, [retrieved: May, 2022]
- [13]] CEF, CEF Digital - eTranslation. [Online]. Available from: <https://ec.europa.eu/cefdigital/wiki/display/CEFDIGITAL/eTranslation>, 2022, [retrieved: May, 2022]
- [14] Django, Django Home Page. [Online]. Available from <https://www.djangoproject.com>, 2022, [retrieved: May, 2022]
- [15] T. Kralidis, pygeoapi Home Page. [Online]. Available from: <https://pygeoapi.io>, 2021, [retrieved: May, 2022]
- [16] OpenLayers, OpenLayers Home Page. [Online]. Available from: <https://openlayers.org>, 2022, [retrieved: May, 2022]
- [17] GeoJSON, GeoJSON Home Page. [Online]. Available from: <https://geojson.org>, 2021, [retrieved: May, 2022]
- [18] GDAL, GDAL/OGR Home Page. [Online]. Available from: <https://gdal.org>, 2021, [retrieved: May, 2022]
- [19] Mapbox, rasterio Home Page. [Online]. Available from: <https://rasterio.readthedocs.io/en/latest/>, 2021, [retrieved: May, 2022]
- [20] INSPIRE, INSPIRE UML-to-GeoJSON encoding rule. [Online]. Available from: <https://github.com/INSPIRE-MIF/2017.2/blob/master/GeoJSON/geojson-encoding-rule.md>, 2019, [retrieved: May, 2022]
- [21] INSPIRE-MIF, INSPIRE Alternative Encoding Model Simplification Rules. [Online]. Available from: <https://github.com/INSPIRE-MIF/2017.2/blob/master/model-transformations/TransformationRules.md>, 2020, [retrieved: May, 2022]
- [22] M. Nordberg, PxWeb API description. [Online] Available from: https://pxnet2.stat.fi/API-description_SCB.pdf, 2020, [retrieved: May, 2022]
- [23] L. Lehto and J. Kähkönen, 2021. OGC API Features HTML-output as a Feature Dashboard, *Abstr. Int. Cartogr. Assoc.*, 3, 176, <https://doi.org/10.5194/ica-abs-3-176-2021>, 2021.
- [24] WhiteBoxTools, WhiteBoxTools Home Page. [Online]. Available from: <https://www.whiteboxgeo.com/geospatial-software/>, 2022, [retrieved: May, 2022]
- [25] CoverageJSON, CoverageJSON Home Page. [Online]. Available from: <https://covjson.org>, 2022, [retrieved: May, 2022]
- [26] SciPy, SciPy Home Page. [Online]. Available from: <https://scipy.org>, 2021, [retrieved: May, 2022]
- [27] three.js, three.js Home Page. [Online]. Available from: <https://threejs.org>, 2021, [retrieved: May, 2022]
- [28] 3DCityDB, 3D City DB Home Page. [Online]. Available from: <https://www.3dcitydb.org/3dcitydb/>, 2022, [retrieved: May, 2022]
- [29] OGC, CityGML Home Page. [Online]. Available from: <https://www.ogc.org/standards/citygml>, 2012, [retrieved: May, 2022]
- [30] CityJSON, CityJSON Home Page. [Online]. Available from: <https://www.cityjson.org>, 2021, [retrieved: May, 2022]
- [31] J. Virtanen, et al., “Near real-time semantic view analysis of 3D city models in web browser”, *ISPRS International Journal of Geo-Information*, vol. 10, no. 3, pp. 1–23, <https://www.mdpi.com/2220-9964/10/3/138>, 2021
- [32] S. Vitalis et al. CITYJSON + WEB = NINJA, *ISPRS Annals of Photogrammetry, Remote Sensing and Spatial Information Sciences*, VI-4/W1-2020, pp. 167–173, <https://doi.org/10.5194/isprs-annals-VI-4-W1-2020-167-2020>, 2020.
- [33] QGIS, QGIS Home Page. [Online]. Available from: <https://www.qgis.org/en/site/>, 2021, [retrieved: May, 2022]
- [34] TUDelft, TU Delft 3D Geoinformation Home Page. [Online]. Available from: <https://3d.bk.tudelft.nl>, 2022, [retrieved: May, 2022]

Decentralised Autonomic Self-Adaptation in a Foraging Robot Swarm

Liam McGuigan, Roy Sterritt, George Wilkie, Glenn Hawe

School of Computing, Faculty of Computing, Engineering and the Built Environment
Ulster University
Jordanstown, N. Ireland

email: mcguigan-l8@ulster.ac.uk, r.sterritt@ulster.ac.uk, fg.wilkie@ulster.ac.uk, gi.hawe@ulster.ac.uk

Abstract— The deployment of a swarm of robots in domains, such as mine clearance or search and rescue operations, requires that they are self-adaptive in order that they may adjust to unforeseen events and up to date information. Much of the research on swarm self-adaptation focuses on the adaptation of individual swarm behaviour, however a top-down approach may allow a swarm to adjust its behaviour on the basis of the combined knowledge of the swarm. This research looks at producing a decentralised autonomic manager to handle such adaptation in a task of foraging robots, by adjusting the range over which robots broadcast help requests based on the perceived density of the swarm. First, the robots are tasked with recognising the initial situation, before responding to two possible events which alter the scenario, namely the destruction of a proportion of the swarm, and a change in the effective communication range. A centralised system is first developed as an idealised system with full swarm knowledge, and then a decentralised version is created to perform the same role on a per-robot basis, using only the information available to it. The performance of the swarm using each autonomic manager is compared against the performance when using a fixed broadcast range identified to be most suitable for the initial circumstances. It is found that both approaches are capable of recognising the initial situation, and of responding to events, however the effectiveness of the response may depend upon additional parameters not taken into account here. The decentralised autonomic manager presented is also found to require the ability to dynamically alter its own parameters in order to be of use.

Keywords- Swarm robotics; Self-adaptation; Autonomic Computing; Simulation.

I. INTRODUCTION

This paper is an extended version of the work published in [1], extending those results and presenting further research.

A swarm of robots, in which the aggregate behaviour of many relatively simple individuals combines to create a more complex set of behaviours [2], can have applications in areas, such as mine clearance [3], search and rescue [4] and space exploration [5][6]. A robot swarm can reduce the demands on any single robot, may accomplish the task more quickly, and can be deployed where sending humans is too dangerous, difficult, or costly.

The ability to self-adapt, that is to adjust behaviour in response to newly acquired information without the need for external guidance, is a requirement of a robotic swarm [7]. Unforeseen events may occur that require adjustment, and factors, such as distance and time, may restrict the ability of

a human operator to act successfully. Self-adaptation can be applied to the swarm in a variety of ways [8], including the development of emergent behaviours [9], evolutionary systems [10] and swarm-level decision making [11].

Autonomic Computing concepts [12][13] can be used for swarm-self adaptation. At the swarm level, an Autonomic Manager (AM) employing a control loop, such as the Monitor, Analyse, Plan, Execute system described by [12] can be used to allow the swarm to assess the current situation and take any action necessary, as seen in Figure 1. This may be implemented in either a centralised manner, with individual robots communicating with a central command unit, or in a decentralised manner with each robot using its own control loop in order to modify its own behaviour in response to shared information and experience.

The objective of this work is to explore the potential for using swarm-level self-adaptation in a swarm of robots to improve performance in a foraging task, specifically the time it takes the swarm to complete the task which may often be an application priority, such as in search and rescue.

Robot swarms are typically decentralised in nature [3], and a similarly decentralised approach for the swarm's self-adaptation is desired. Initially, a centralised approach is used as an exploratory stage to determine if an AM provides any benefits. This is followed by an implementation of a decentralised approach to performing the same self-adaptation. As the centralised AM exists only to explore self-adaptation options, it will not take a more active role in coordinating the swarm in its task.

Each approach to the AM aims to achieve performance improvement through the modification of the range at which individual robots communicate with their neighbours for

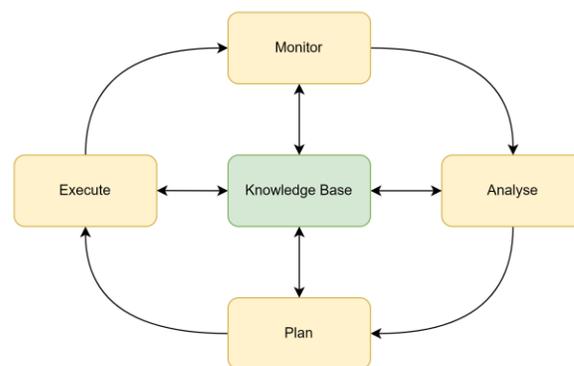


Figure 1. MAPE-K loop, as used by an autonomic component. The loop proceeds through each of the four stages in turn.

assistance. The swarm is tasked with deciding the appropriate communication range, and then two unforeseen events are introduced. The first, robot destruction, tests the swarm's ability to react to the sudden change in swarm size, such as a loss of robots in a search and rescue task due to the hazardous environment. The second, a change to communications quality, represents a situation where the ability of the robots to communicate with each other may be hampered by a change in environmental conditions.

The rest of this paper is structured as follows. Section II discusses related work in swarm-level adaptation. Section III describes the simulation and the foraging task used. Section IV explains how the two approaches to autonomic management have been applied to the swarm, while Section V describes the test scenarios. Section VI reports the results and explores the implications, and Section VII concludes the paper with a summary, and any future research directions.

II. RELATED WORK

The location where adaptation is applied to a swarm is important when considering the intended goal. Much of the research in adaptation focuses on the level of the individual agent, where the resulting swarm performance is affected by the aggregate of these individual behaviours [8]. This level of adaptation can have a dramatic impact on performance, but it is difficult for any single robot to take advantage of information that is only available when viewing the larger picture, or to make decisions affecting the behaviour of other members of the swarm, such as cooperation or communication.

Adaptation at the swarm level can counter some of these problems. [14] describes an approach to moderating the size of the swarm in order to reduce degraded performance due to congestion. Robots keep track of the conflicts that occur when two robots attempt to occupy the same cell. If the number of conflicts crosses a threshold, virtual pheromones can be deposited at the entrance in order to instruct robots to leave or join the area. Hence, the swarm can adjust its size based on the combination of each robot's collision tracking data.

In [15], a group of unmanned aerial vehicles (UAVs) are patrolling an area defined by a set of cells, with the aim of ensuring that cells are visited often enough during the mission. Individual UAVs decide their next target on the basis of values assigned to the cells by a central system based on UAV visitation. Different strategies for assigning those values are explored, and so the central system becomes an effective behaviour adaptation method for the group.

As discussed in Section I, autonomic concepts may be used for swarm self-adaptation. [16] describes an adaptation pattern in which one robot in the swarm takes on the role of an AM, running a control loop with visibility of the whole system. In the case study presented, the swarm was tasked with exploring an unknown area. Robots communicate their positional and explorational information with the AM, which can direct them to underexplored areas. Recognising that a centralised system may be a bottleneck, a decentralised variant is also used in which the robots share

the information with their neighbours. Both approaches perform much better than a basic pheromone-based approach.

A partially distributed approach described in [17] uses a group of UAVs, together with communication base stations taking on the role of AMs, engaged in a search task. If one of the UAVs leaves the active area and loses the communication link, the base stations are able to recognise the failure and reposition themselves in order to retrieve the UAV, while also minimising disruption to the rest of the swarm.

In a previous paper [8], cooperation strategies for swarms were investigated to determine the potential for using an AM to select between them based on the situation. Here, we build on that research by using the Help Recruitment strategy, exploring the potential for using an AM to modify the broadcast range parameter in order to improve performance, and then creating a decentralised implementation to achieve that.

III. SIMULATION OF FORAGING ROBOTS

This research employs a time-stepped simulation of a heterogeneous swarm of agents, engaged in a variant of a foraging task, as in a previous work [8]. The following subsections describe the simulation and task, the behaviour of the robots, and how communications and energy are handled by the simulation.

A. Simulation and Foraging Task

The simulation presents a world consisting of a rectangular grid, in which several items and robots are placed at random. Each item or robot may be one of two possible types, represented by their colours, as shown in Figure 2. A single cell may contain only one item, however it can contain any number of robots. Each cell can therefore be considered to represent an area much larger than the footprint of a single robot, and thus the simulation may ignore potential collisions between robots.

The simulation is updated in a time-stepped manner.

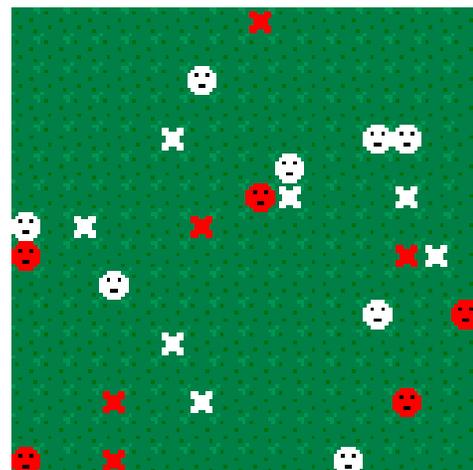


Figure 2. A portion of the world state during a simulation. The colour of a robot (face) or item (cross) indicates its type.

Each tick of the update loop, all robots are updated in turn, for a fixed time delta. Hence, when measuring the time taken for the swarm to complete a task, it is the number of ticks of the simulation that must be measured, rather than the real time taken for the simulation to complete, which may be impacted by factors such as the simulation playback speed, host platform, and so forth.

The robots are given the task of foraging the items, with each robot capable only of foraging an item that matches its own type and occupies the same cell. The task ends once all items have been successfully foraged. Each item is foraged in the position found, with no requirement to return the item to a home base – this process is therefore analogous to applications, such as the analysis of mineral deposits, or environmental cleanup.

B. Robot Behaviour

The behaviour of the robots follows the Help Recruitment strategy developed in [8]. Figure 3 is a state machine diagram showing the states and transitions employed by the robots in this strategy.

A robot begins in the Explore state. Each tick of the simulation, the robot first checks for an item in the current cell, and if none are found, it selects a random adjacent cell to move into. If an item is found, the robot will transition to the Forage state.

In the Forage state, the robot first checks the item to see if it matches the robot's type. If it does, the item is successfully foraged and the robot resumes exploration. However, if the type does not match, the robot broadcasts an

initial help message to neighbouring robots, stating that it has found an item of a given type, and requires assistance from suitable robots. After sending the message, it moves to the Wait For Offers state.

A robot remains in the Wait For Offers state for two ticks, listening for responses from other robots. After this period, the robot will select the offer from the nearest robot and send an assignment message to them, before resuming exploration. If no offers have been received, exploration is resumed with no assignment.

A robot receiving a help request will only respond to it if it matches the type, and is in the Explore state. In this case, the robot sends a offer message to the robot requesting help, and moves to the Wait For Assignment state, where it can remain for up to three simulation ticks before resuming exploration. If it receives an assignment message in this time, it then moves to the Respond state.

In the Respond state, the robot moves directly towards the location of the item to be foraged, but continues to check the cells it passes through, sending help requests if necessary. The robot continues responding until it arrives at the target destination, where it can forage the item if it is still present, or resume exploration if another robot has foraged the item ahead of it.

In this way, robots are able to cooperate. Where a robot is unable to forage an item it finds, it can recruit a nearby robot to carry out the task instead. As reported in [8], engaging in this cooperation increases the performance of the swarm compared to having no cooperation in place.

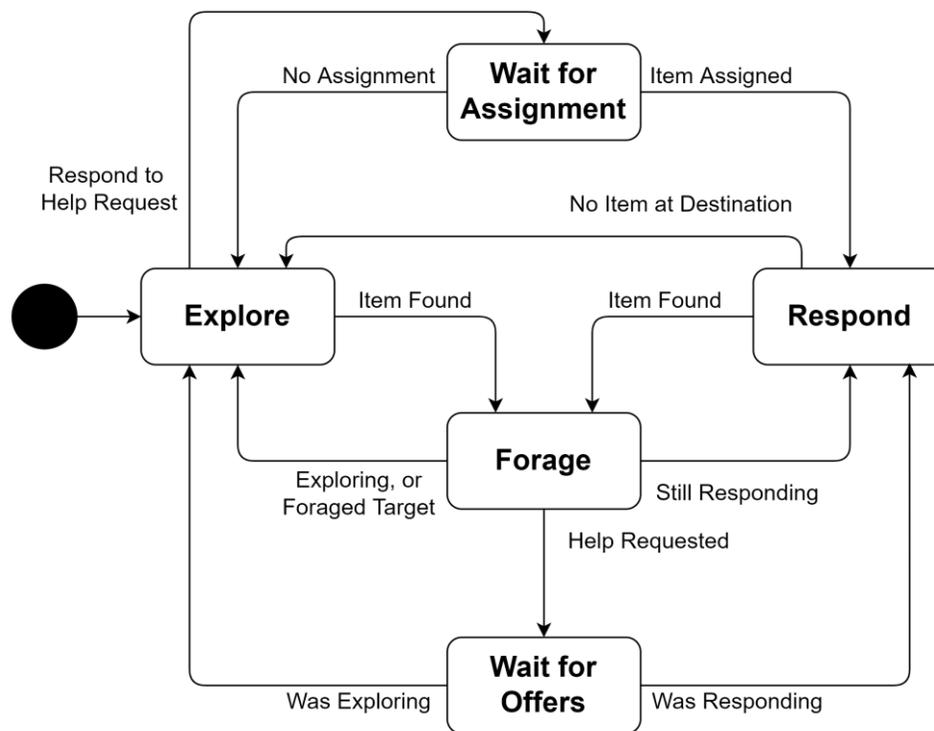


Figure 3. State Machine for the Help Recruitment cooperation strategy.

C. Communication and Energy

Communication within the simulation works by queuing each message, and processing all messages at the end of each simulation tick. Each message has a point of origin and a range, and this is used to ensure that it is sent to all robots within range. Each robot stores the received messages in a list, and will process them at the beginning of their next simulation tick, but before doing so it will shuffle the list in order to avoid the update order of robots affecting the underlying behaviour.

To explain why this is necessary, consider a robot requesting help, and receiving a response from two robots the same distance away. If the list of received messages is not shuffled, the first of these robots to be updated by the simulation would always be the robot selected to assist. When the message list is shuffled, the second robot has an equal chance of being selected.

Communications within the simulation are further affected by a global quality setting, represented as a percentage. The effective range of each message is then determined by multiplying the message's intended broadcast range by this percentage expressed as a value between 0 and 1, hence a value of 50% results in a message only reaching half the distance it intended. This is a very simple approach, however it allows for the conditions faced by the robots to change in a way that requires recognition and action.

The energy expended by each robot is measured by assigning each action a cost, in arbitrary units. Each robot has an upkeep cost of 1 unit per tick, which is incurred in addition to the costs of actions taken. Foraging an item costs 1 unit, while movement costs 1 unit per cell moved, or 1.41 units when moving diagonally. The energy cost of communications depends on the maximum range of the broadcast, prior to the effects of the global communications quality setting. This is applied using the power law stated in (1), where r is the range of the broadcast in cells.

$$cost = 0.01 \times r^2 \quad (1)$$

Measuring the energy expended by the robots is included as a means of exploring the potential impacts of using increased broadcast ranges within the swarm.

IV. AUTONOMIC ROBOTS

When robots require assistance, they send a help message with a broadcast range set prior to the mission. However, as robots that receive a message will pause awaiting assignment, messages that reach a higher number of potential helpers may have a negative impact on the performance of the swarm. In addition, as the energy costs of broadcasting for help increase exponentially with distance, shorter broadcasts are preferred.

A preliminary study was carried out to determine best performing broadcast range for each robot density. The foraging task was conducted using a variety of swarm sizes and broadcast ranges, and selecting the best performing range for each swarm size. The density of each robot type was calculated as in (2),

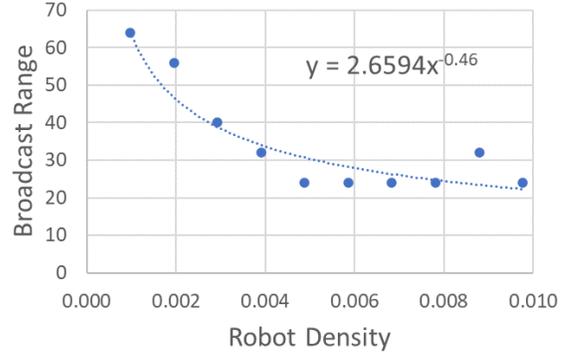


Figure 4. Derivation of the ideal broadcast range function. The points indicate the best performing range for the given density, based on mean ticks to completion.

$$\delta = r / A, \quad (2)$$

where δ is the density of robots of a given type, r is the number of robots of that type, and A is the area of the map in square cells. Using this information allowed a plot of ideal broadcast range against density, as shown in Figure 4. Fitting an approximate trend line to the plot leads to an equation for determining the broadcast range to use based on the density of the smallest group of robots, as in (3).

$$range = 2.6594 \times \delta_{min}^{-0.46} \quad (3)$$

The broadcast range used by the robots in the swarm may be set prior to the mission based on the known densities and operating area, however if an event occurs which changes these parameters mid-mission, or the parameters are not known ahead of time, then it is necessary to give the swarm the ability to manage the broadcast range itself.

This has been accomplished in two ways, employing either a centralised or a decentralised system. The following subsections describe how these approaches work.

A. Centralised Autonomic Manager

In the centralised approach, each robot sends a pulse message to a Central Autonomic Manager (CAM) every 16 simulation ticks, containing its type, exploration information, and the maximum distance from which it received messages from neighbours since the last pulse.

With the same period, the CAM can use the data to determine the composition of the swarm, estimate the area of the map, and detect any changes in communications quality.

Composition is determined simply by counting the number of received messages from each type of robot since the last period.

The area of the map is calculated by having each robot keep track of the rectangular bounds within which it has thus far explored. As each is received by the CAM, it updates its own rectangular bound, increasing it in size on each axis as needed to contain the latest information. The

area of this rectangle will then approximate the area of the map within which the robots operate.

To detect changes in communications quality, the robots are required to send out local pulse messages with a range of 8 cells. Each pulse contains the position of the robot that sent it. Between each update to the CAM, a robot keeps track of the maximum distance of a received pulse, and that information is sent in its message to the CAM. The CAM similarly tracks the maximum pulse range received. As a randomly scattered swarm may expect to receive pulses up to the expected range of 8 cells, a lower value can be used to indicate a drop in communications quality, with the perceived quality being calculated as in (4), where $range_{max}$ is the maximum received pulse range, and the denominator of 8 is the broadcast range of any given pulse.

$$quality = range_{max} / 8 \quad (4)$$

The CAM uses the information obtained each period to calculate the density of each robot type within the world, and thus the ideal broadcast range using (3). This is then divided by the perceived communications quality in order to compensate, or if the quality is zero, the CAM can instruct the robots to avoid sending help requests to avoid wasted energy and time.

B. Decentralised Autonomic Manager

In the decentralised approach, each robot has a Decentralised Autonomic Manager (DAM), which only sends a local pulse every 32 ticks with a fixed broadcast range of 8 cells, containing the sending robot type. Between each pulse, a robot updates a count for each robot type from which it receives a pulse message.

For each period, the robot calculates a density value as in (2), with the area used being that of a circle with the same radius as the pulse messages. If this density is non-zero, it is then used to calculate an ideal broadcast range using (3), which the robot uses for any subsequent help requests. However, if no messages were received, the calculated density is zero, so the robot sets the broadcast range at twice the pulse range, reasoning that no robots are within the pulse range, but may potentially lie just outside it.

The DAM does not attempt to determine the communications quality. Instead, it is assumed that if the quality drops, the number of received messages from other robots may also drop, resulting in a lower density of robots in the local area and a corresponding increase in attempted broadcast range.

It can be noted that this approach does not take into account swarm-level knowledge, and is instead localised adaptation based on the knowledge available to individual robots. However, there is some degree of knowledge about the swarm as each robot is aware of the composition of their local neighbourhood. The focus here is on implementing a method by which the robots can correctly adjust their broadcast range without relying on any central system.

V. TEST SCENARIOS

In determining the viability of the two autonomic approaches, three sets of tests were conducted for each of

autonomic approach. First, the performance of the swarm was measured in a set of fixed scenarios. Second, the ability of the AMs to react to a sudden drop in the number of robots in the swarm was tested. The third test was to determine the AMs' abilities to react to a change in communications quality.

Each of the tests was conducted in a 128x128 map, seeded with 256 items which were equally distributed between red and white types. Each of the setups within a test scenario was run 100 times to obtain a sample of results, and the performance of the swarm was measured based on the simulation ticks taken to successfully forage all items. In addition to performance data, the energy costs for the entire swarm have been measured to determine the relative efficiency by which the robots complete the task.

The following subsections describe each of the test scenarios.

A. Autonomic Manager Performance

To test the hypothesis that the autonomic managers used are capable of determining a suitable broadcast range for the swarm and perform no worse than the appropriate fixed range for a given swarm density, tests were performed comparing the swarm using a set fixed broadcast ranges, and then using each autonomic manager.

The fixed broadcast ranges used were 4, 8, 16, 24, 32, 40, 48, 56 and 64 cells. The tests were repeated with 64, 128 and 256 robots, always equally distributed between the two types.

The mean time taken by the swarm over the 100 runs was then compared across each test setup, with the best performing fixed broadcast range identified, and then compared to both the centralised and decentralised autonomic approaches.

B. Robot Destruction

To test the ability of the autonomic managers to recognise a sudden change in the swarm composition, an event was set up to occur after 300 simulation ticks, in which an equal number of robots of each type are removed from the simulation. This has the effect of decreasing the density of the swarm, which the AMs should be able to detect and react accordingly.

The test with 256 robots was run in four scenarios, with the percentage of robots destroyed in each scenario being 25%, 50%, 75% and finally 90%. These tests were carried out using the best performing fixed broadcast range identified in the autonomic manager performance tests described previously, and also with each AM.

The mean time taken by the swarm over the 100 runs was then compared across each test setup, and also compared to the corresponding fixed broadcast range test with no robot destruction, so it may be seen how the robot destruction affects performance, and the impact of using an autonomic manager.

C. Communications Quality Change

To test the ability of the autonomic managers to react to a change in the communications quality, an event was set up

to occur after 300 simulation ticks, in which the communications quality was changed from its initial setting to another value. This has the effect of either reducing or increasing the effective broadcast range of the robots, and the AMs should be able to adjust.

The test with 256 robots was run in four scenarios, with the communications quality changes being 100%-25%, 25%-100%, 100%-0% and 0%-100%.

As in the robot destruction scenario, these tests were carried out using the best performing fixed broadcast range identified in the AM Performance scenario, and then with each AM.

The mean time taken by the swarm over the 100 runs was then compared across each test setup, and also compared to the corresponding fixed broadcast range test without any change in communications quality, so it may be seen how the change affects performance, and the impact of using an autonomic manager.

VI. RESULTS

The following subsections discuss the results of the three main test scenarios, followed by an additional focus on the DAM pulse period, and an overall summary.

A. Autonomic Manager Performance

Figure 5 shows the simulation ticks taken for task completion by swarms of 64, 128 and 256 robots respectively, and the energy costs for the 256-robot swarm. Independent t-tests were performed between the identified best broadcast range for each swarm size, against the performances of both the CAM and DAM, and the results of this are summarised in Table I.

It can be seen that the CAM is capable in each case of performing as well as the best performing fixed broadcast range, completing the task in a similar time. It can also be seen that while the performance of the swarm does not appear to degrade with higher broadcast ranges, the energy cost does increase, and so using a CAM may be more efficient than simply selecting a high broadcast range. However, it should be noted that the cost of communications between a CAM and individual robots has not been factored in during this work.

The results show that the DAM with a pulse range of 8 cells does not perform as well when the swarm size is low. The pulse messages sent between robots to facilitate the DAM approach will be subject to the same performance degradation as the help broadcast messages, in lower robot densities.

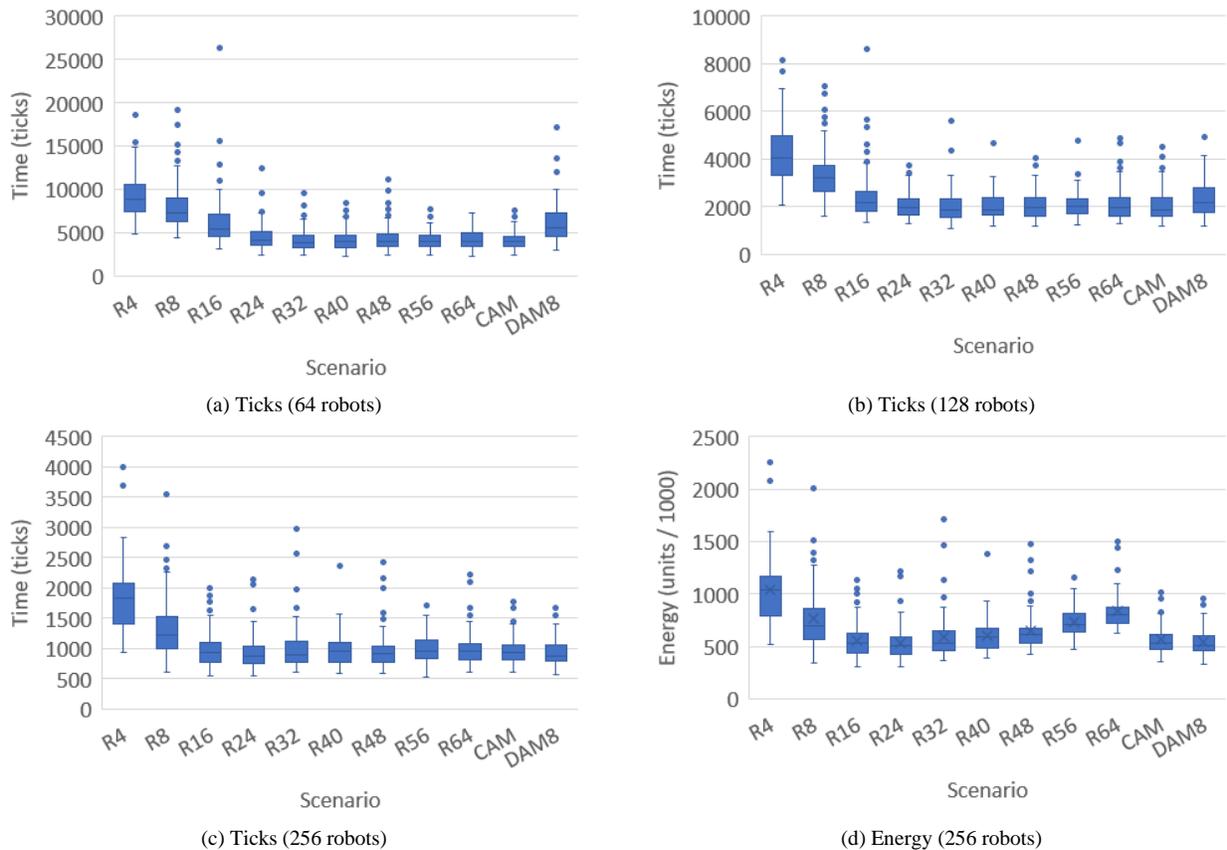
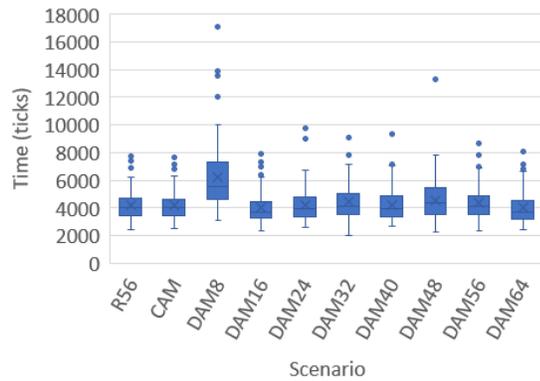


Figure 5. Ticks taken for a each swarm of robots to complete the task (a-c), and energy usage for the 256-robot swarm (d), for each broadcast range tested, and the two AMs. Circles represent outliers in the data.

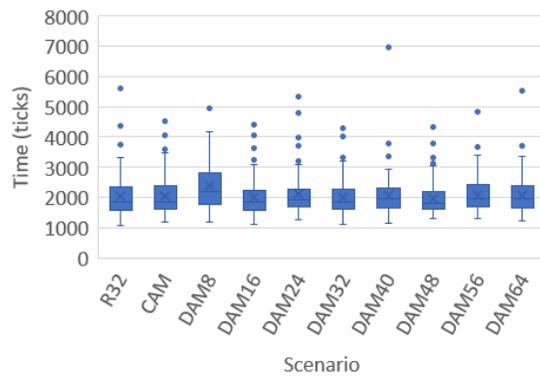
TABLE I. AM PERFORMANCE T-TEST RESULTS

Swarm Size	Ideal Range	Fixed Range		CAM		Deg. Of Freedom	t-statistic	p-value
		Mean	SD	Mean	SD			
64	56	4151.59	1031.553	4153.54	1044.386	198	-0.013	0.989
128	40	2022.26	564.917	2024.05	596.051	198	-0.022	0.983
256	24	936.90	267.334	983.77	248.969	198	-1.283	0.201
DAM8								
64	56	4151.59	1031.553	6230.53	2347.897	135.847	-8.107	0.000
128	40	2022.26	564.917	2382.46	809.580	176.932	-3.649	0.000
256	24	936.90	267.334	948.82	250.759	198	-0.325	0.745
DAM16								
64	56	4151.59	1031.553	4024.84	1152.987	198	0.819	0.414
128	40	2022.26	564.917	1987.78	611.588	198	0.414	0.679
256	24	936.90	267.334	945.80	258.645	198	-0.239	0.811

Figure 6 shows the performance of swarms of 64 and 128 robots, with a variety of DAM configurations with pulse ranges set between 8 and 64 cells. These results show that an increase of the pulse range to 16 cells will correct this problem for these swarm sizes. Table I includes the results of t-tests comparing the performance of a fixed broadcast range to the DAM with pulse ranges of 8 and 16 cells.



(a) 64 robots



(b) 128 robots

Figure 6. Ticks taken for each swarm to complete the task, comparing performance of the DAM with various pulse ranges. Circles represent outliers in the data.

This suggests that some means of adjusting the pulse range dynamically based on the local robot densities may be useful to allow the swarm to not only adapt its help broadcast range, but also its own autonomic processes.

B. Robot Destruction

Figure 7 shows the performance of the swarm for the robot destruction scenario, for 25%, 50%, 75% and 90% of robots destroyed. Independent t-tests were performed between the identified best broadcast range from the previous subsection, and the performances of both the CAM and DAM, and the results of this are summarised in Table II.

Comparing the performance of the swarm in the two fixed-range scenarios shows that the destruction of robots has a detrimental impact on the swarm's performance, as can be expected as reducing the size of the swarm means fewer robots are left to complete the same task.

The results show that for the cases where 25% and 50% of robots are destroyed, the CAM performs as well as the fixed broadcast range but confers no advantages to the swarm. However, as the number of robots destroyed increases, the CAM starts to show its worth, completing the task faster in both the 75% and 90% destruction scenarios. Throughout, the energy cost correlates with the performance.

As more robots are destroyed, it appears that the CAM's ability to increase the help broadcast range, and therefore increase the chances of remaining robots successfully signalling for other robots to assist in foraging an item, results in an improvement in the swarm's performance over retaining the original fixed broadcast range.

As with the AM Performance tests, the results for the DAM show that a pulse range of just 8 cells does not allow it to improve performance, and in fact it performs worse than even a fixed broadcast range. Figure 8 shows the performance of the swarm using a variety of DAMs with pulse ranges set from 8 to 64 cells, and compares against the fixed broadcast range and the CAM. It can be seen that increasing the pulse range to 24 cells would produce similar

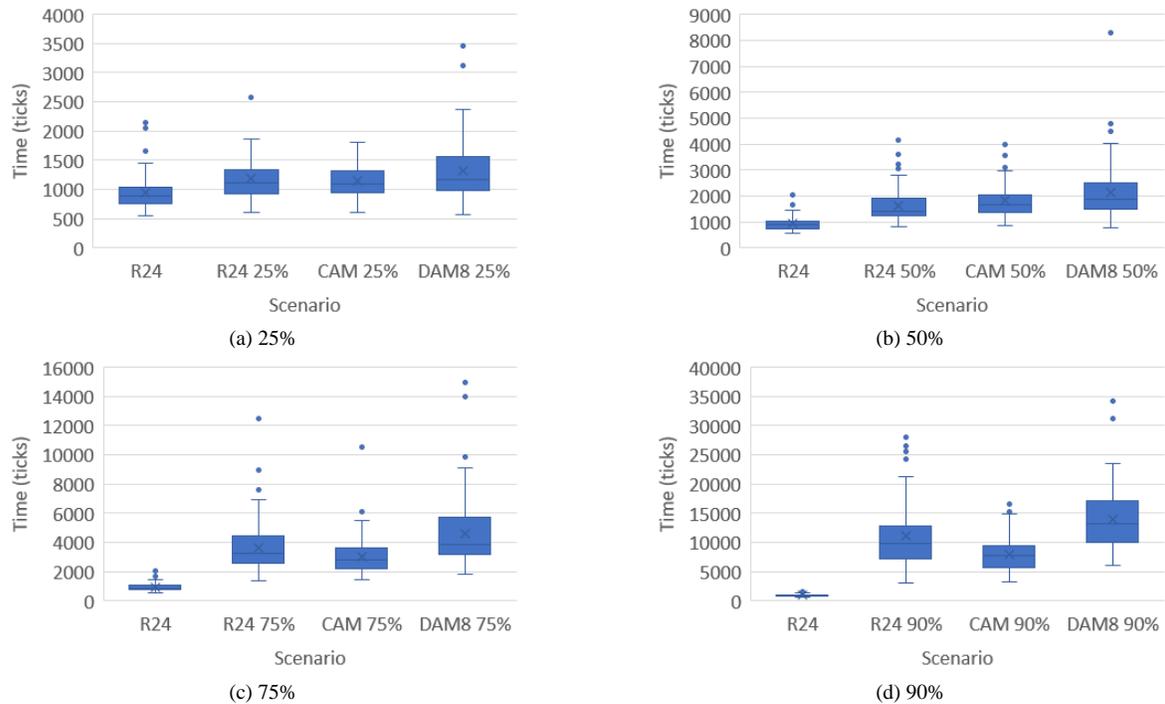


Figure 7. Ticks taken by the swarm to complete the task during the Robot Destruction scenario. Circles represent outliers in the data.

performance to the CAM, again highlighting the need for the ability to dynamically change this value.

C. Communications Quality Change

Figure 9 shows the performance of the swarm in the communications quality change scenarios. Independent t-tests were run comparing the fixed range performance with that where the AM is active, and the results are summarised in Table III. The equivalent tests comparing energy usage are shown in Table IV.

It is only when the communications quality drops from 100% to 25% that the CAM confers any advantage to the swarm, performing better than the fixed broadcast range, and performing as well as the swarm without any change in communications quality, suggesting it is successful in

counteracting the reduction in effective broadcast range. It can also be seen that despite an effective four-fold increase in the broadcast range it uses, which amounts to a 16-fold increase in the cost of each broadcast, the energy requirements are still lower than using a fixed broadcast range.

In the other scenarios, we do not see any benefit to using a CAM. In the 25%-100% change, the CAM is actually less efficient than using a fixed broadcast range, despite taking the same amount of time to complete the task. To explain this, it is likely that during the first 300 ticks of the simulation, where the communications quality is reduced to 25%, there are much more items in the world for robots to find, meaning a much higher chance of an item being found by a robot of the wrong type, and therefore many more help

TABLE II. ROBOT DESTRUCTION T-TEST RESULTS

Destroyed Robots / %	R24		CAM		Deg. of Freedom	t-statistic	p-value
	Mean	Std. Dev.	Mean	Std. Dev.			
25	1187.51	342.602	1150.36	303.995	198	0.811	0.418
50	1630.59	617.813	1811.13	606.876	198	-2.085	0.038
75	3664.05	1709.930	3041.41	1226.075	179.516	2.959	0.004
90	11196.18	5458.347	7974.65	2753.011	146.305	5.270	0.000
			DAM8				
90	11196.18	5458.347	13930.47	5191.608	198	-3.630	0.000
			DAM24				
90	11196.18	5458.347	7831.72	2820.210	148.341	5.476	0.000

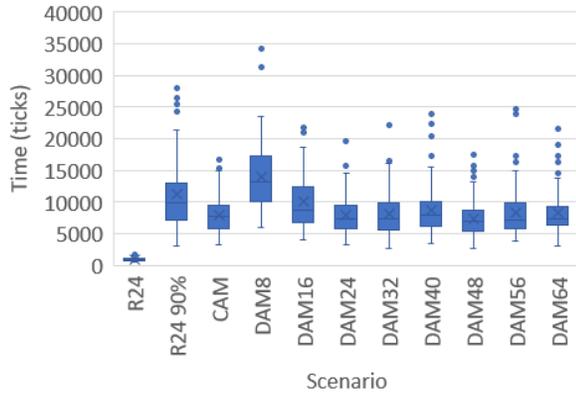


Figure 8. Ticks taken for the swarm during the 90% Robot Destruction scenario, for each DAM configuration. Circles represent outliers in the data.

requests being sent. In the 100%-25% scenario, there are fewer items remaining in the world by the time the communications quality drops, and so fewer help requests are sent.

In the cases where the communications quality begins or ends at 0%, no statistical differences can be seen between

the CAM and the fixed broadcast range. This is likely because at 0% communications quality, no cooperation is possible, and the performance of the swarm is dominated by the random search for items.

Focusing on the 100%-25% scenario for the DAM, we again see that the 8 cell pulse range is detrimental to the performance of the swarm. Figure 10 shows the performance of the swarm with a variety of DAMs with pulse ranges set between 8 and 64 cells. It can be seen that a pulse range of 40 cells performs best in this scenario. As with the other test cases, this highlights the need for a dynamic pulse range.

D. Decentralised AM Pulse Period

The above results show that adjusting the broadcast range of the DAM can have an impact on the performance of the swarm. In light of this, it is necessary to investigate the effects of the only other parameter in the DAM, that of the period between pulses.

To test this, a swarm of 128 robots, each equipped with a DAM set to a broadcast range of 8 cells, was tested in the standard performance scenario, using periods of 8, 16, 24, 32, 40, 48, 56 and 64 ticks. A one-way ANOVA test was used to determine if any statistical difference exists between the resulting sets of data using the period as the independent

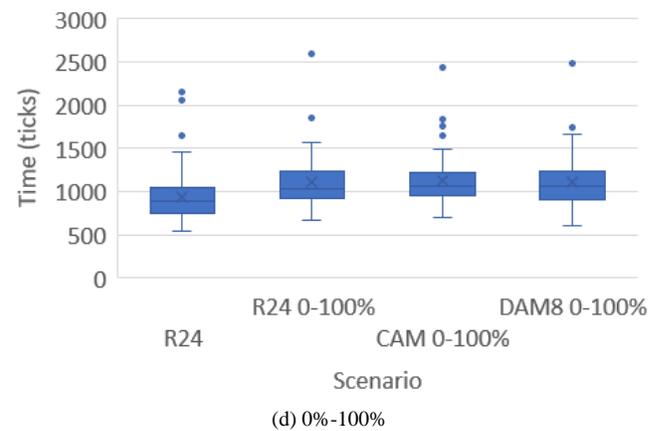
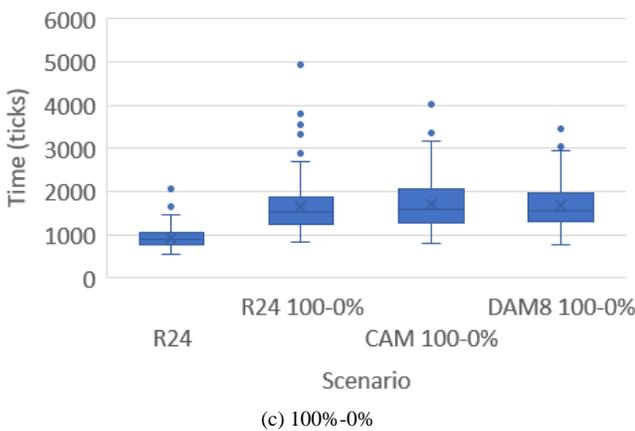
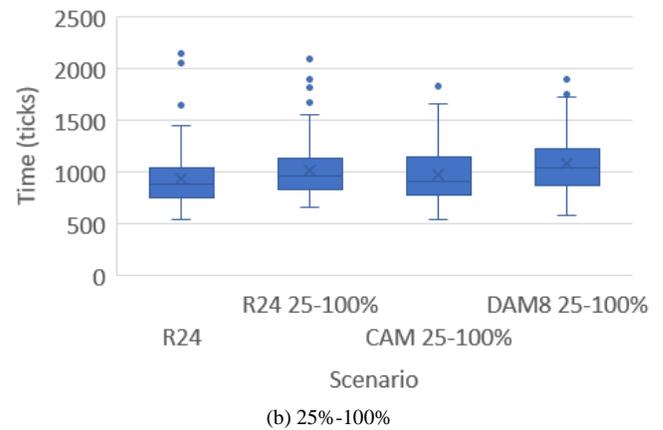
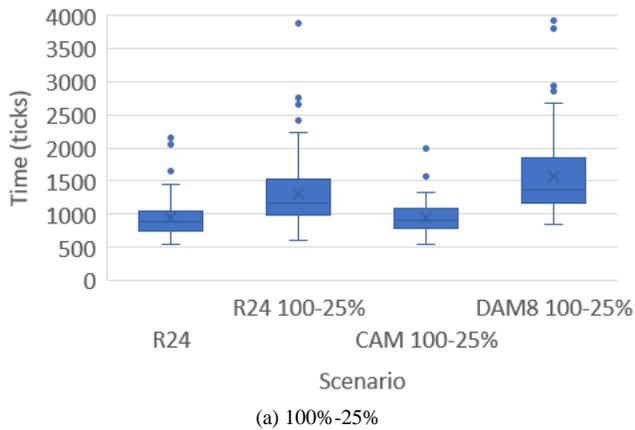


Figure 9. Ticks taken by the swarm to complete the task during the Communications Quality Change scenario. Circles represent outliers in the data.

TABLE III. COMMUNICATIONS QUALITY T-TEST RESULTS (TICKS)

Quality Change	R24		CAM		Deg. of Freedom	t-statistic	p-value
	Mean	Std. Dev.	Mean	Std. Dev.			
100 – 25%	1314.86	518.360	944.96	239.680	139.481	6.477	0.000
25 – 100%	1015.27	268.303	980.89	286.952	198	0.875	0.383
100 – 0%	1663.61	706.816	1724.27	625.697	198	-0.643	0.521
0 – 100%	1108.01	276.859	1132.90	302.442	198	-0.607	0.545
DAM8							
100 – 25%	1314.86	518.360	1572.25	612.837	198	-3.207	0.002
DAM32							
100 – 25%	1314.86	518.360	1031.65	358.278	198	4.494	0.000
DAM40							
100 – 25%	1314.86	518.360	947.32	254.206	143.958	6.367	0.000

TABLE IV. COMMUNICATIONS QUALITY T-TEST RESULTS (ENERGY)

Quality Change	R24 / 1000		CAM / 1000		Deg. of Freedom	t-statistic	p-value
	Mean	Std. Dev.	Mean	Std. Dev.			
100 – 25%	746.49	292.797	563.98	136.668	140.184	5.648	0.000
25 – 100%	574.95	151.535	730.59	163.040	198	-6.993	0.000
100 – 0%	943.32	399.203	986.42	356.575	198	-0.805	0.422
0 – 100%	627.33	156.363	646.28	172.379	198	-0.814	0.417
DAM8							
100 – 25%	746.49	292.797	899.07	349.310	198	-3.348	0.001
DAM32							
100 – 25%	746.49	292.797	687.40	231.925	198	1.582	0.115
DAM40							
100 – 25%	746.49	292.797	684.42	176.740	162.689	1.815	0.071

value.

Figure 11 shows the performance and energy cost of the swarm in completing this task, and the one-way ANOVA test showed there is no statistically significant difference in the data ($F(7, 792) = [0.578]$, $p = 0.774$).

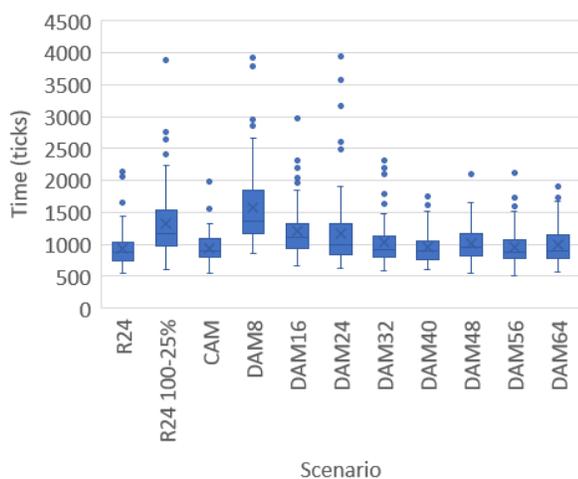


Figure 10. Ticks taken for the swarm during the 100%-25% communications quality scenario, for each DAM configuration. Circles represent outliers in the data.

As such, the period with which the pulse broadcasts are sent does not appear to have any impact on the performance of the DAM. However, further tests may be useful to explore the impact of the period when using other broadcast ranges or swarm sizes.

E. Summary

The results above show that the presence of an Autonomic Manager can have benefits for the performance of the swarm, however it is possible for the AM to reduce performance in some circumstances, and these will require further investigation to determine what other parameters may be affecting the swarm’s performance. If the AM can be developed to account for these further variables, it may be able to counter their effects.

For example, if estimates of the density of items in the world can be made by the AM, this can be used to reduce the communication range when the item density is high, avoiding interruptions that occur during the help broadcasts which may lead to poorer performance.

It can also be seen that the idealised Centralised Autonomic Manager produces a performance that can be replicated by a decentralised approach, without the problems associated with having a central bottleneck for the swarm. However, the pulse range used by the system for allowing robots to detect neighbours needs to be dynamic for the DAM to account for unknowns either in the initial

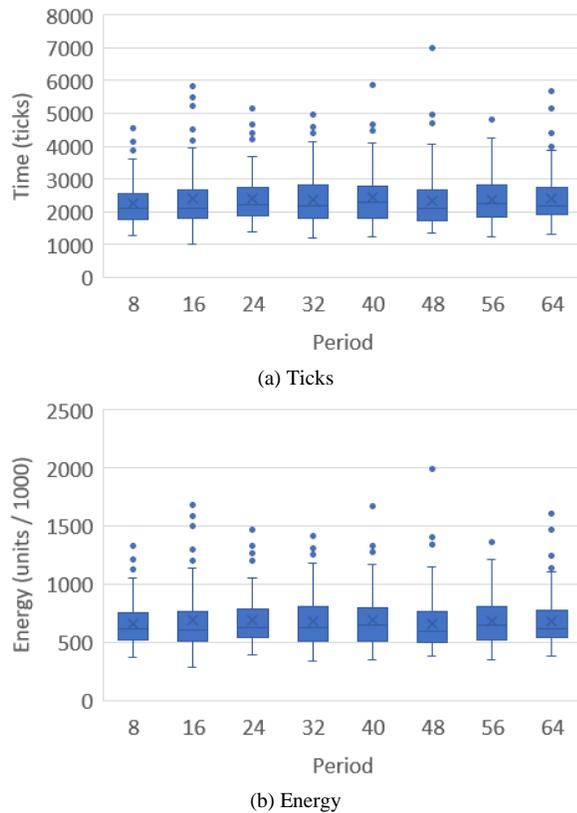


Figure 11. Ticks taken (a) and energy cost (b) of a swarm of 128 robots, using a DAM8 with a variety of pulse periods. Circles represent outliers in the data.

situation, or that arise through events that occur during the mission.

VII. CONCLUSION AND FUTURE WORK

This research used a simulation of a robotic swarm equipped with two different autonomic management capabilities, able to manage the performance of the swarm through adjustment of the intra-swarm communication range.

A centralised approach was used as an idealised situation in which a central system has full knowledge of the swarm with which to act, as a means of investigating the potential for improving performance through the use of an AM. Further to this, a decentralised approach was developed that allowed individual robots to monitor their own environment and select an appropriate broadcast range without requiring any central component.

The findings show that a centralised AM is capable of finding an appropriate broadcast range when given a task where the map size and number of robots in the swarm is not initially known to the AM, and must be deduced from information gathered by the individual robots.

When a robot destruction event occurs, the centralised AM proves beneficial to the swarm when the robot loss is high, capable of completing the task faster than using a fixed broadcast range. No benefit is seen when the robot loss is low.

In the event of a change in communications quality, the centralised AM is capable of improving performance when the quality drops from high to low without dropping out entirely, but not when the quality starts low and increases. This is likely due to the increased item density during the early stages of the task, and it is worth exploring this factor to see how the AM might measure and take item density into account.

A centralised system has problems that have not been replicated directly in this work, such as the potential for the central AM to be a bottleneck on performance, the need and energy cost required for individual robots to maintain a link to the central AM, and reduced autonomy of any one robot. To counter this, a fully decentralised approach was developed and tested.

The decentralised approach was found to be capable of matching the performance of the centralised system, despite robots not having complete knowledge of the swarm, achieved only by tracking local pulse messages sent by neighbouring robots to estimate the swarm density. The findings however show that a means of dynamically adjusting the range of these pulse messages is required for the decentralised AM to reach the desired performance, while the pulse period does not appear to have an effect.

Future work will investigate methods to allow the dynamic adjustment of the pulse range, as well as exploring other scenarios in which a decentralised autonomic manager may be of use to a swarm, and the possibility of sharing decisions made by individual robots in order to help guide other robots, thus restoring the concept of swarm-level knowledge that is not present in the current decentralised approach. Additional work may also explore other situations that may affect performance, such as more complex maps containing obstacles, differing distributions of robot types, more complexity in the foraging task, on-board batteries that drain and require recharging, and further events that may occur to unexpectedly change the world state.

REFERENCES

- [1] L. McGuigan, R. Sterritt, and G. Wilkie, 'Centralised Autonomic Self-Adaptation in a Foraging Robot Swarm', in *The Seventeenth International Conference on Autonomic and Autonomous Systems (ICAS 2021) IARIA*, May 2021, pp. 11–17.
- [2] H. Hamann, *Swarm Robotics: A Formal Approach*. Springer International Publishing, Cham, Switzerland, 2018.
- [3] I. Navarro and F. Matía, 'An Introduction to Swarm Robotics', *ISRN Robot.*, vol. 2013, pp. 1–10, 2013.
- [4] L. Bayindir, 'A review of swarm robotics tasks', *Neurocomputing*, vol. 172, pp. 292–321, Jan. 2016.
- [5] R. Sterritt, G. Wilkie, C. Saunders, M. D. C. Gama, G. Hawe, and L. McGuigan, 'Inspiration for Space 2.0 from Autonomic-ANTS (Autonomous NanoTechnology Swarms) Concept missions', presented at the Reinventing Space Conference, 2019.
- [6] A. Farahani, G. Cabri, and E. Nazemi, 'Self-* properties in collective adaptive systems', in *Proceedings of the 2016 ACM International Joint Conference on Pervasive and Ubiquitous Computing: Adjunct*, Heidelberg, Germany, Sep. 2016, pp. 1309–1314.

- [7] G. Beni, 'From Swarm Intelligence to Swarm Robotics', in *Swarm Robotics*, Berlin, Heidelberg, 2005, pp. 1–9.
- [8] L. McGuigan, C. Saunders, R. Sterritt, and G. Wilkie, 'Cooperation Strategies for Swarms of Collaborating Robots: Analysis of Time-Stepped and Multi-Threaded Simulations', *Int. J. Adv. Syst. Meas.*, vol. 14, no. 3 & 4, pp. 44–58, 2021.
- [9] J. Prasetyo, G. De Masi, and E. Ferrante, 'Collective decision making in dynamic environments', *Swarm Intell.*, vol. 13, no. 3, pp. 217–243, Dec. 2019.
- [10] N. Bredeche, E. Haasdijk, and A. Prieto, 'Embodied evolution in collective robotics: A review', *Front. Robot. AI*, vol. 5, p. 12, 2018.
- [11] C. Saunders, R. Sterritt, and G. Wilkie, 'Collective Communication Strategies for Space Exploration', *J. Br. Interplanet. Soc.*, vol. 72, no. 12, pp. 416–430, 2019.
- [12] J. O. Kephart and D. M. Chess, 'The vision of autonomic computing', *Computer*, vol. 36, no. 1, pp. 41–50, Jan. 2003.
- [13] E. Vassev, R. Sterritt, C. Rouff, and M. Hinchey, 'Swarm Technology at NASA: Building Resilient Systems', *IT Prof.*, vol. 14, no. 2, pp. 36–42, Mar. 2012.
- [14] J. Zelenka, T. Kasanický, and I. Budinská, 'A Self-adapting Method for 3D Environment Exploration Inspired by Swarm Behaviour', in *Advances in Service and Industrial Robotics*, Cham, 2018, pp. 493–502.
- [15] K. S. Kappel, T. M. Cabreira, J. L. Marins, L. B. de Brisolará, and P. R. Ferreira, 'Strategies for Patrolling Missions with Multiple UAVs', *J. Intell. Robot. Syst.*, vol. 99, pp. 499–515, Sep. 2019.
- [16] M. Puviani, G. Cabri, and F. Zambonelli, 'Agent-based Simulations of Patterns for Self-adaptive Systems', in *Proceedings of the 6th International Conference on Agents and Artificial Intelligence*, ESEO, Angers, Loire Valley, France, 2014, pp. 190–200.
- [17] M. Allison, M. Robinson, and G. Rusin, 'An Autonomic Model-Driven Architecture to Support Runtime Adaptation in Swarm Behavior', in *Advances in Information and Communication*, Cham, 2020, pp. 422–437.

Interpretation Support by Extracting Time Series Classification Patterns using HMM from Text-based Deep Learning

Masayuki Ando

R-GIRO, Ritsumeikan University

Kusatsu, Japan

email: oh23mandou@ec.usp.ac.jp

Yoshinobu Kawahara

Institute of Mathematics for Industry, Kyushu University, and

RIKEN Center for Advanced Intelligence Project

Fukuoka, Japan

email: yoshinobu.kawahara@riken.jp

Wataru Sunayama

School of Engineering, The University of Shiga Prefecture

Hikone, Japan

email: sunayama.w@e.usp.ac.jp

Yuji Hatanaka

Faculty of Science and Technology, Oita University

Oita, Japan

email: hatanaka-yuji@oita-u.ac.jp

Abstract—We developed an interpretation support system for classification patterns extracted from deep learning with texts using a hidden Markov model (HMM) and verified its effectiveness. It is well known that classification patterns by using deep learning models are often difficult to interpret the reasons derived. In the proposed system, the content of deep learning results is extracted using structure of HMM, and classification patterns are provided for the system users to interpret the learned features. The system then displays learned network structures so that anyone can easily understand the learning results. In verification experiments to confirm the effectiveness of the system, based on the learning result of deep learning classifying sentences, participants were divided into two groups. One group used the proposed system, while the other group used a system that displays words with high Term Frequency-Inverse Document Frequency (TFIDF) values. Both groups were instructed to give meanings of classification patterns peculiar to each output. The results indicate that the participants who used the proposed system were able to understand the meanings of the classification patterns of deep learning with texts better than those who used the comparison system.

Index Terms—*interpretation support; deep learning; text mining; text classification; data visualization*

I. INTRODUCTION

The applications of artificial intelligence (AI) systems based on deep learning have been rapidly increasing, including image recognition, automatic driving of automobiles, automatic delivery of packages using drones, and assistance in medical diagnoses.

In the U.S., there are unmanned convenience stores that allow customers to accurately identify and pay for products simply by holding them in their hands and leaving the store without going through a cash register. A major Japanese pharmaceutical company is also collaborating with an AI research institute in the U.K. to search for new compounds

using AI in an attempt to greatly improve the efficiency of drug development.

There is, however, a problem with deep learning in that the criteria for prediction and classification by learning are unknown and incomprehensible to humans. This problem is especially serious in fields such as medicine and automated driving, where the reliability and safety of learning results are important. In text processing, if it is possible for humans to understand the decision criteria of deep learning, new applications of deep learning are expected such as understanding how to write good electronic medical records on the basis of the differences between electronic medical records written by newcomers and veterans or obtaining information that can be used as hints for product development by analyzing questionnaires and reviews.

We focused on the text classification problem and considered a method to extract classification patterns including time-series information from deep learning using the likelihood calculation method of hidden Markov model (HMM). The ‘classification pattern’ is the feature set that contributes to the classification, and is a clue to understanding the basis for the classification criteria. This method has already been treated in previous research [1] published in ACHI 2021, which has shown a certain level of effectiveness. However, prior research did not sufficiently explain the conversion of RNN weights to HMM probability distributions, and subjects’ interpreted sentences were not analyzed as a result of the experiment. Therefore, this paper describes the proposed method, including the explanation of the normalization of RNN weights, and the results of the analysis of the interpreted sentences produced by the subjects using the proposed system.

The proposed method calculates the likelihood of the classification patterns (i.e., an evaluation value indicating the importance of the classification patterns) from the trained

deep learning, just as an HMM calculates the likelihood of a given observation series. Then, we build a system for users to interpret the highly rated classification patterns. Interpretation of this classification patterns allows users to understand the rationale for the classification criteria. We believe that by constructing a system that enables even novice data analysts to interpret classification patterns, we can create an environment in which users of cloud-based machine learning application programming interfaces and individuals who wish to carry out simple text mining can easily interpret the learning results.

We provide support for interpreting classification patterns as one approach to understanding the basis for classification criteria in deep learning. Therefore, rather than improving the accuracy of classification results, we focus on how humans can understand the basis of classification criteria. In addition, this approach does not provide the users with a mechanical determination of the basis for the classification criteria. It will only provide assistance to the users in finding the basis for the classification criteria. This is because if the basis of the classification criteria is judged mechanically, a new problem arises as to whether the judgment is correct.

In this study, we considered the order in which words appear in a sentence (i.e., time-series information) to be important for understanding the basis of the classification criteria. Therefore, for preprocessing words to be learned in deep learning, we use the one-hot vector format, in which each node in the input layer has a one-to-one correspondence with a word, and for deep learning models, we use recurrent neural networks (RNNs), which can also learn time series information of words. The proposed system can also be applied to long short-term memory (LSTM) and gated recurrent units, which are extensions of RNNs. However, in this case, the proposed system does not use the gate information in the cell. An HMM [2] that uses neural networks to calculate transition probabilities has also been proposed, but it does not have a framework for interpreting the training results of RNNs. The proposed system extracts classification patterns from RNN training results by referring to the likelihood calculation method of HMM.

We discuss related work in Section II. In Section III, we describe the configuration and details of our HMM-based classification-pattern interpretation support system in deep learning networks. In Section IV, we describe the experiments we conducted to verify the effectiveness of the proposed system and conclude the paper in Section V.

II. RELATED RESEARCH

AI (in this context, we refer primarily to systems that use deep learning) has been playing an increasingly important role in a wide variety of situations, such as medical treatment, image judgment in automated driving systems, and automated stock trading. With the advent of cloud-based AI [3], it has become possible to use AI easily even on personal mobile devices.

There is, however, a black box problem in deep learning. Deep learning learns information through a very complex process and can make predictions and classifications with high

accuracy. However, due to the complexity of the process, it is very difficult for humans to explain the decision criteria of deep learning.

Explainable AI (XAI) [4] has been gaining attention as a research field that focuses on explaining the reliability and fairness of deep learning models and understanding the decision criteria. Research on XAI began with the need to explain what has been learned to understand and trust the behavior of deep learning models [5] [6]. In fact, research has been conducted to try to explain the behavior of the model itself, such as attempting to explain the behavior on the basis of the correlation between the data and variables in the model [7], and using counterfactual conditional statements to help users understand the behavior of the model [8]. In addition to interpreting model behavior, there are also studies that focused on the stability and reliability of model behavior, such as countermeasures against malicious data [9] or evaluating model behavior and stability by applying model behavior to different logic circuits or decision trees [10] [11].

If we look at XAI research in terms of its objectives, we find that there is a large amount of research in what is called informational systems [12] [13]. An informational system is a basic method in XAI research, in which additional information is added to the output of the model, and the user can infer the validity and correctness of the AI's answer. In image processing, a method [14], [15] has been proposed for emphasizing the parts of the input image that contribute to the output. In natural language processing, however, it is difficult to apply methods used in the image processing field directly. Merely highlighting a part of the input text, as with the method called attention [16], is considered insufficient as an explanation of the classification criteria since it remains unclear what type of learning is going on inside the deep learning process.

Our aim was to develop a system to support the interpretation of classification criteria on the basis of the classification patterns including the time-series information of words by using an RNN as a deep learning model, taking the text classification problem as an example. In addition, this research exists as one of the XAI approaches, but as mentioned in the introduction, it does not provide a machine-determined basis for deep learning classification criteria. It extracts classification patterns as clues for understanding the basis of the classification criteria and encourages users to interpret them.

III. INTERPRETATION SUPPORT SYSTEM FOR CLASSIFICATION PATTERNS FROM DEEP LEARNING NETWORKS USING HMM

In this section, we describe the configuration and details of our system for supporting the interpretation of classification patterns using HMMs in deep learning networks for text-based classification tasks.

A. System Configuration

The configuration of the proposed system is shown in Figure 1. A set of texts with correct labels is used as training data,

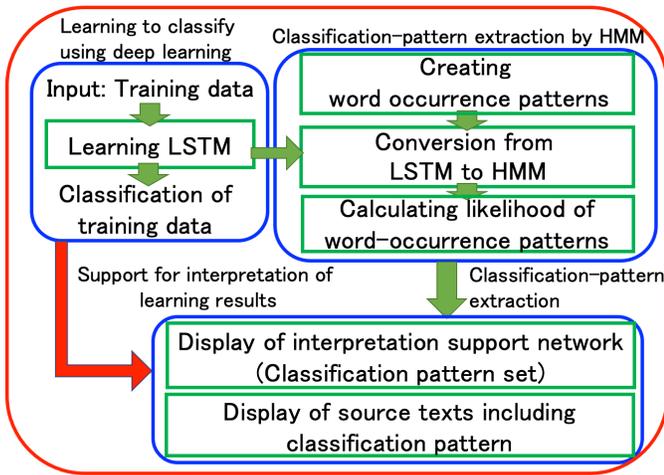


Fig. 1: System configuration

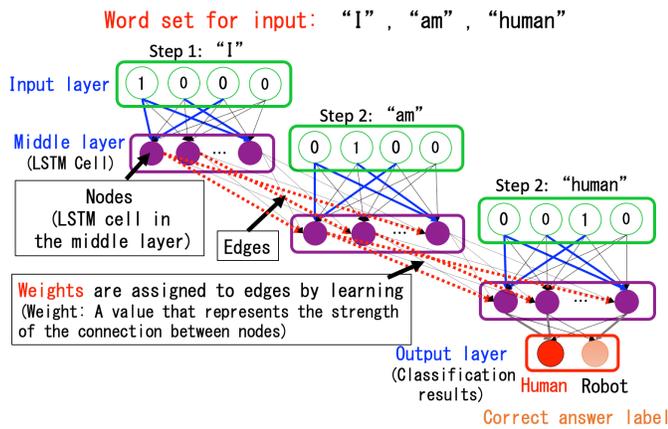


Fig. 2: Example of weighting by using RNN

and an RNN is used to classify them. The trained weighted network is then converted into an HMM, and the likelihood of word-occurrence patterns in the text set (source text) used for training is calculated. Finally, the word-occurrence patterns with the highest likelihood are displayed on the interface as classification patterns, and the user interprets the classification patterns. At this time, the user can arbitrarily set the number of classification patterns to be displayed. The system also has a source-text-display function that allows the user to refer to the source text to better understand the meaning of the classification patterns.

B. Training an RNN

We used RNNs, which are generally used to learn ordering patterns in time series data, and considered situations in which they are applied to the problem of classifying text sets (Figure 2). In Figure 2, an RNN with one intermediate layer is used as an example. In an RNN, words are trained in turn at each

time step, and the information is saved in the middle layer. At this time, the nodes that fire in the middle layer change with each time step, and this change corresponds to a change in state in the HMM. The reason for using an RNN is that we considered not only the type and frequency of words but also the time series of word occurrences to be important features in sentences. Also, unlike most deep learning research, we did not aim to achieve high classification accuracy but to build a system that encourages interpretation of deep learning networks, which are generally difficult to interpret.

In RNNs, a set of texts with correct labels is used as input, and the word vectors of each text (nouns, verbs, and adjectives in the text are represented by 0 and 1, respectively) are used to learn the edge weights in an RNN with one intermediate layer so that the classification accuracy is high. In this study, we extracted the weight set from the learned weighted network and applied it to an HMM to extract the classification patterns.

To interpret the learned classification patterns, it is assumed that proper training has been carried out. For this reason, we assume that the network has been trained by deep learning so that the classification accuracy of the test data in the 10-fold crossover test during training or the test data different from the training dataset is at least 90% and that the network does not contain large errors.

C. Creating Word-occurrence Patterns

To improve the interpretability of the classification patterns by making them closer to the actual text, the proposed system uses the word patterns that actually appear in the text set used for training the RNN as the observation series to be fed to the RNN converted to HMM (see Section III-D for details). In this case, all word patterns that satisfy the following conditions are used as candidates. The length (number of words) of the word-occurrence patterns can be set arbitrarily by the user. However, the length of each pattern cannot be set individually.

- The words in a word pattern are the nouns, verbs, and adjectives in the source text (adjectives may be omitted in experiments).
- The words in the word-occurrence pattern are only those words that appear in at least 1% of the sentences frequency.
- The order of words in a word pattern should be based on the actual order of words in the source text.

The reason for these conditions is that we aimed to promote the interpretation of patterns that are typical among classification patterns (i.e., those with a large amount of applicable data). Therefore, even if we extract classification patterns consisting of particles or infrequently used words, it is difficult to enable interpretation of typical patterns. There is also a possibility of misinterpretation when interpreting patterns of word sequences that do not appear in the source text. To solve this problem, we use nouns, verbs, and adjectives as word-occurrence patterns, and only words with sentence frequencies above a certain threshold. The order of the words in word-occurrence patterns is also based on the time series of the actual words.

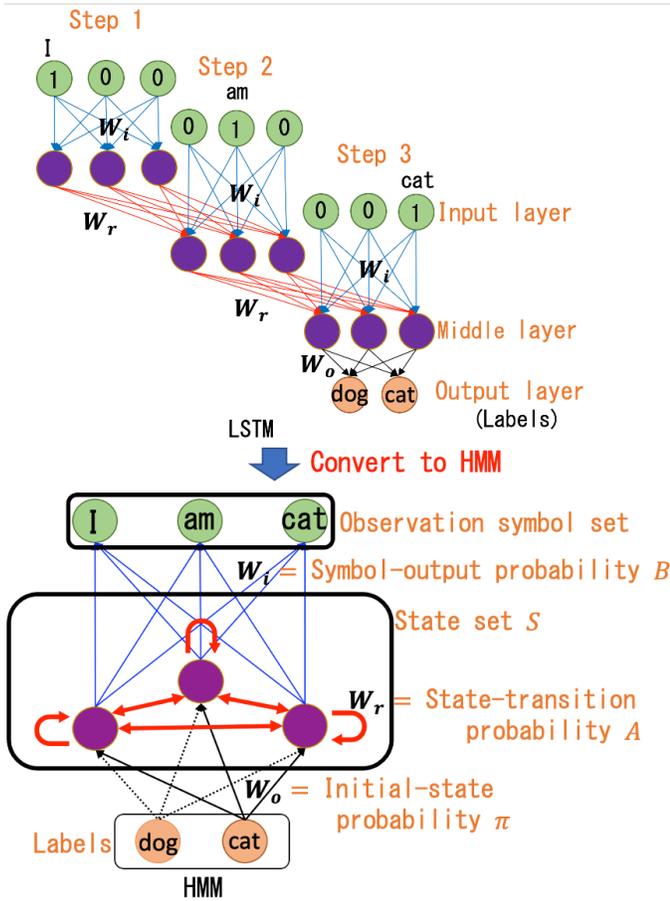


Fig. 3: Interaction between RNN and HMM

D. Conversion of RNN to HMM

An HMM is a non-deterministic finite state automaton model with two processes: a state and an observation symbol (output). When the state is stochastic, the observation symbol is output in a stochastic manner. An HMM can calculate the likelihood, which is the value of how plausible the change in the observation symbol is. Therefore, by applying an RNN to an HMM, we can express how much a certain word-occurrence pattern contributes to the output of the RNN in terms of likelihood. Therefore, by transforming the RNN into an HMM, we can express how much a given word occurrence pattern contributes to the output of the RNN in terms of likelihood. Although we use the word ‘transform’ here, we are not actually changing the structure of the RNN. We are fitting the components of the RNN to the HMM so that the likelihood calculation method of the HMM can be applied to the RNN.

In the proposed system, the weighted network obtained by training an RNN is transformed into an HMM to estimate the likelihood of a word-occurrence pattern (Figure 3). The reason for using HMM is that, as shown in Figure 4, there are similarities in the structure of RNN and HMM, and we believe that the method of calculating the likelihood of HMM can be

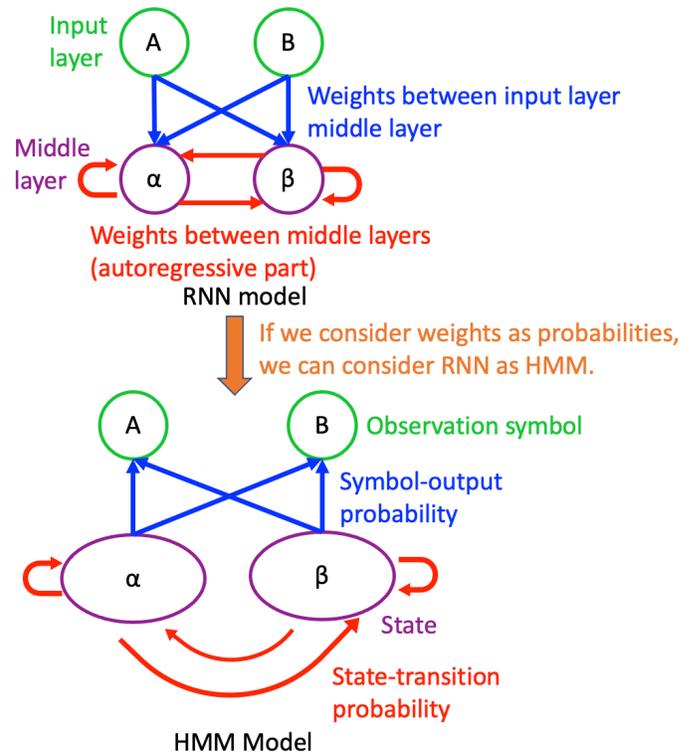


Fig. 4: Comparison of RNN and HMM

applied to RNN as well. The input layer node of the RNN is the observation symbol set of the HMM, and the middle layer node is the state set S . Similarly, the weight set W_r between the time series of the intermediate layer (by recursive processing) is the state-transition probability A , and the weight set W_i between the intermediate layers of the input layer is the symbol-output probability B . Let the set of weights between the middle and output layers W_o be the initial state probability π (π depends on the destination to be selected at that time).

However, the weight set of the RNN does not satisfy the condition of probability. Therefore, for the weights $w_i (1 \leq i \leq N)$ that make up the weight vector w between certain intermediate layers (N is the number of elements in w), if the weights are negative, they are set to 0 and w'_i (Equation (1)), and the value w'_i is normalized so that the sum of the weights is 1 (Equation (2)).

$$w'_i = \max\{0, w_i\} \quad (1)$$

$$w''_i = \frac{w'_i}{\sum_{s=1}^N w'_s} \quad (2)$$

From the above, we can treat the weight set of the RNN as the A and B of the HMM by normalizing each weight of the weight vector to sum up to 1 using Equation (2) for the weight set between each layer of the RNN.

TABLE I: Examples of extracted classification patterns

Likelihood Rank	Extracted Classification Patterns
1st	“Fresh cream” → “Frozen” → “Potato starch”
2nd	“Fresh cream” → “Potato starch” → “Frozen”
3rd	“Strawberries” → “White bean jam” → “Potato starch”
4th	“Brush” → “Potato starch” → “Frozen”
5th	“Chin” → “White bean jam” → “Strawberry”

E. Likelihood Estimation of Word-occurrence Patterns from Trained Weighted Networks Using HMM

This section describes the calculation of the likelihood of the set of word-occurrence patterns created in Section III-C. For the RNN weighted network converted to HMM in Section III-D, the observation sequence (the word-occurrence pattern described above) is input to $\mathbf{O} = \{o_1, o_2, \dots, o_T\}$ (T is the length of the observation sequence, i.e., the length of the word-occurrence pattern), and the number of states (the number of intermediate layer nodes) is N (the state number is i, j), \mathbf{A} is given by Equation (3), \mathbf{B} is given by Equation (4), and $\boldsymbol{\pi}$ is given by Equation (5).

$$\mathbf{A} = \{a_{ij} | a_{ij} = P(s_{t+1} = j | s_t = i)\} (1 \leq i, j \leq N) \quad (3)$$

$$\mathbf{B} = \{b_{ij}(o_t) | b_{ij}(o_t) = P(o_t | s_{t-1} = i, s_t = j)\} \\ (1 \leq i, j \leq N, 1 \leq t \leq T) \quad (4)$$

$$\boldsymbol{\pi} = \{\pi_i | \pi_i = P(s_0 = i)\} (1 \leq i \leq N) \quad (5)$$

When there is a word-occurrence pattern \mathbf{O} for a destination x , the initial state probability is denoted as $\boldsymbol{\pi}_x$, and the likelihood $P(\mathbf{O} | \boldsymbol{\pi}_x, \mathbf{A}, \mathbf{B})$ is calculated using the following equation.

$$P(\mathbf{O} | \boldsymbol{\pi}_x, \mathbf{A}, \mathbf{B}) = \sum_{all S} P(\mathbf{S} | \boldsymbol{\pi}_x, \mathbf{A}, \mathbf{B}) P(\mathbf{O} | \mathbf{S}, \boldsymbol{\pi}_x, \mathbf{A}, \mathbf{B}) \\ = \sum_{all s_0 \dots s_T} \pi_{x s_0} a_{s_0 s_1} b_{s_0 s_1}(o_1) \cdot a_{s_1 s_2} b_{s_1 s_2}(o_2) \cdot \\ \dots \cdot a_{s_{T-1} s_T} b_{s_{T-1} s_T}(o_T) \quad (6)$$

Finally, the likelihood is calculated for all word-occurrence patterns using Equation (6), and the word-occurrence patterns are extracted as classification patterns that contribute to classification in the order of increasing likelihood.

F. Interpretation-support-network Display

The extracted set of classification patterns in the previous section, which are strongly connected to the classification destination, is displayed as an interpretation support network with the proposed system. In this network, words are displayed as orange nodes (① in the Figure 5) and the time-series relationships between words are displayed as blue arrows (②)

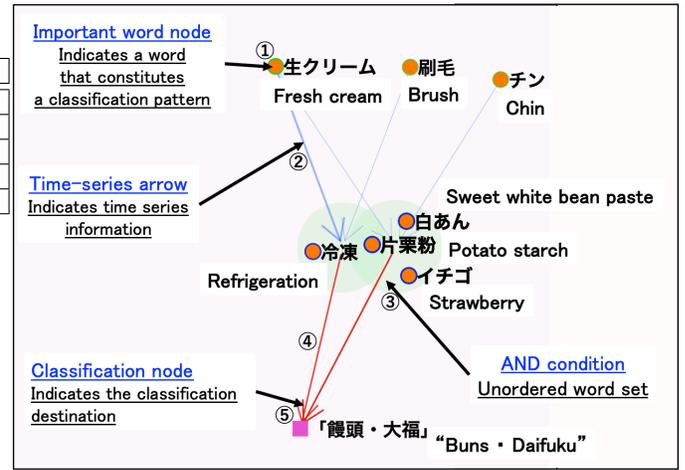


Fig. 5: Example of displayed interpretation support network

between nodes to make it easier to understand the words and the time-series relationships between words in the classification pattern. The magnitude of the likelihood is indicated by the thickness of the arrows. Furthermore, nodes with arrows in both directions are considered to have a weak time-series relationship and displayed as a single group in the green area (③). To indicate which classification pattern belongs to which destination, a red arrow (④) connecting the purple node (⑤) that displays the destination name and the last word node of the classification pattern is displayed.

The interpretation support network displayed from a set of texts on how to make five types of Japanese sweets (collected from Cookpad [17]) is shown as an example in Figure 5. The user first selects the node at the bottom of the interface where the name of the classifier (in this case, “Buns and Daifuku”) is displayed for interpretation. The system first extracts the classification patterns for the selected destination name in the user’s desired number in the order of likelihood. The extracted classification patterns are shown in Table I. Next, an interpretation support network is displayed, with the words of the extracted classification patterns as nodes and the time-series relations between the words as arrows. Finally, by looking at the interpretation support network, the user can find out what words and time-series relationships between words contribute to the selected classification destination and interpret the patterns. At this time, the user can use interpretation support functions such as displaying and examining any classification pattern from a group of classification patterns, and a source text display function that displays the meaning of words in the classification pattern (details are described below).

It is important to note that the classification patterns of the selected classifiers shown in the interpretation support network are only the characteristics of the selected classifiers compared with other classifiers and not the general characteristics of the word. For example, in the example shown in Figure 5, if you find the pattern “Fresh cream” → “Frozen” and “Potato

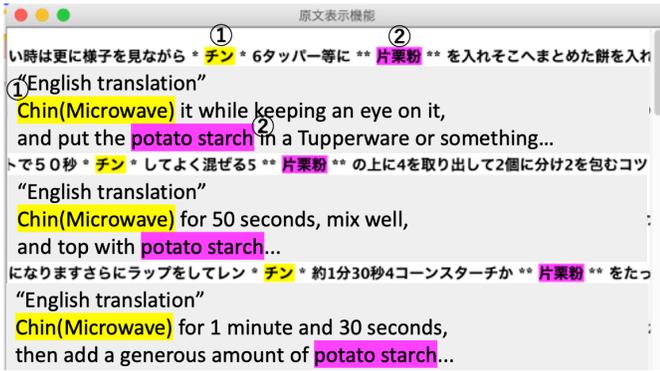


Fig. 6: Example of the source text (selecting the words “Chin” and “Potato starch” for the text about “Buns and Daifuku”)

starch”, you can interpret that the difference from the other four Japanese sweets is the way of making the “Buns and Daifuku”, such as “mixing fresh cream with frozen potato starch” or “mixing fresh cream with potato starch and freezing”. Finally, by interpreting each Japanese sweets classification pattern in the same way, one can understand the basis of the classification criteria for classifying the five Japanese sweets based on the interpretation.

G. Function for Displaying Source Text of Classification Patterns

To interpret the classification patterns, it is difficult to understand the actual context in which the words were used from the word information alone. For this reason, the source-text-display function shows how the words in the classification pattern are actually used in the text used for training.

By selecting a word (clicking on the node) on the interpretation support network, the user can see the sentence that contains the word in the source text. Selected words are highlighted. However, for ease of viewing, we limit the number of words displayed to ten before and ten after the selected word per sentence. Up to two types of words can be selected, in which case all sentences between the words are displayed. The order in which the “selected words” occur in the displayed sentence is based on the order in which the user selected the words. Figure 6 shows an example of the source-text display of the classification pattern for five different ways of making Japanese sweets when the text “Buns and Daifuku” is used as the classification destination and the words “Chin (meaning the sound of a microwave signaling it has finished cooking in Japanese)” (① in the Figure 6) and “Potato starch” (②) are selected in order.

IV. EXPERIMENT TO VERIFY EFFECTIVENESS OF PROPOSED SYSTEM

In this section, we describe the experiment we conducted to verify whether participants without extensive knowledge of deep learning can interpret the classification patterns on the basis of the word-occurrence patterns output with the proposed system.

A. Experimental Procedure

The experiment consisted of the three tasks listed in Table II, in which the participants were asked to interpret the classification patterns of sentences classified into the “output labels” specified for each task. To make the interpretation easier for the participants and facilitate the analysis of the interpretation results, we set an interpretation objective for each task. The “Tsundere” described in Table II refers to a girl who is cold and demanding at first meeting or in public, but sometimes shows kindness. In addition, “Deredere”, discussed below, refers to girls who show sweet or tender feelings toward a specific person throughout. Other details of the data used for the task are described in Section IV-B. The experiment was conducted involving 16 undergraduate and graduate students who had no extensive knowledge of deep learning, and they were divided into two groups: one using the proposed system and the other using a comparison system. The experiment was conducted without changing the members of the group because we focused only on the validity of the interpretations given by the participants to the source-text data without asking about the quality of the interpretations, which may be greatly influenced by the participants’ personalities and ways of thinking.

We used a system that extracts words specific to a specified output label by the Term Frequency-Inverse Document Frequency (TFIDF) value of the expression as the comparison system. We compared the difference between the interpretation based on the characteristic words and their combinations with the comparison system and the interpretation based on the time series of the words with the proposed system.

We asked the group using the proposed system to find words that contribute to classification (one word, combinations, and time series) using the proposed system. We asked the group using the comparison system to find the words that contributed to the classification by looking at a list of words arranged in order of TFIDF value. The TFIDF value i of a word in a source text is obtained as shown in Equation (7), where i is the word in the text. In addition, the source-text-display function can be used in the comparison system.

While participants may have prior knowledge of the text being tested, when interpreting, participants should consider whether the interpretation actually applies to the source text rather than their own knowledge. Therefore, we believed that the presence or absence of prior knowledge would have little effect on the experimental results.

$$\text{TFIDF}_i = \text{sentence frequency for word } i \times \left(\log \left(\frac{\text{output labels num}}{\text{DF value for word } i} \right) + 1 \right) \quad (7)$$

The following steps of the experimental procedure were done by both groups. The number of classification patterns displayed with the proposed system was set to five, consisting of three words, in order of increasing likelihood. The number of words displayed with the comparison system was set to 15 to match the proposed system.

TABLE II: Experimental tasks given to participants and interpretive objectives

Title	Content	Purpose of Interpretation
Task 1 “Character dialogues”: Output label “Tsundere”	Classify the lines of characters in anime and manga with unique characteristics: Ask the students to interpret the characteristics of the lines of characters with “Tsundere” characteristics.	Assuming you are a novelist, find a pattern of word usage specific to the “Tsundere” character for your novel and give your interpretation of it.
Task 2 “Consumer electronics reviews”: Output label “useful”	Classification of reviews about popular consumer electronics on Amazon: Ask students to interpret the characteristics of reviews with a large number of “this review was useful”.	Assuming you are a reporter introducing home appliances, find the patterns of word usage specific to “helpful reviews” about popular home appliances and give your interpretation of them.
Task 3 “Game reviews”: Output label “useful”	Categorize reviews of popular game software on Amazon: Ask students to interpret the characteristics of reviews with a large number of “this review was useful”.	Assuming that you are a reporter introducing a game software, find the pattern of word usage specific to “helpful reviews” and give your interpretation of it.

TABLE III: Deep learning data for each task

	Character dialogues	Consumer electronics reviews	Game reviews
Number of study texts	1500	3108	4419
Number of characters per text (average)	40	244	455
Input layer nodes	510	916	1809
Intermediate layer nodes	10	10	15
Output layer nodes	3	3	3
Classification accuracy	98.7%	99.2%	96.7%

Step 1 Select the output labels to be interpreted: In the “Character dialogues” task, we targeted the lines of characters classified as “Tsundere”. For the “Consumer electronics reviews” and “Game reviews” tasks, we included reviews with a rating of 4 or higher and a “Usefulness” rating of 10 or higher.

Step 2 Read the “Purpose of Interpretation” corresponding to each selected output label to understand its content.

Step 3 For the selected output, display the “Interpretation support network” and find ten features (one word, combinations, time series order, etc.) that may contribute to the output.

Step 4 Ask the user to devise an interpretation of the features in accordance with the “Purpose of Interpretation” using the source-text-display function.

B. Details of Experimental Data and Deep Learning Model Used

Table III lists the deep learning data for each task used in this experiment. For the task “Character dialogues,” we used a total of 1,500 dialogues with the characteristics of “Tsundere,” “Deredere,” and “Normal” characters, 500 each from the “Tsundere bot,” “Deredere bot,” and “Normal Character dialogues bot” on Twitter. For each Twitter bot, we used the top bot accounts (data-acquisition date: July 10, 2020) when we searched for “Tsundere Twitter bot,” “Deredere Twitter bot,” and “Character dialogues Twitter bot”. For the task “Consumer electronics reviews,” we used a total of 3108 reviews from the top 50 “popular consumer electronics” on Amazon [18]: 1036 each of “Useful” (4 stars or more and 10 or more “Useful

people”), “Useless” (4 stars or more and 0 “Useful people”), and “Low-rated” (2 stars or less) reviews. The reason the “4 stars or more” reviews were used was that it was thought that there were some meaningful reviews and some not so meaningful reviews among the same high-rated reviews, and it was intended to give an interpretation of the results of learning to distinguish them. In the “Game reviews” task, we used a total of 4419 reviews from the top 100 “Popular game software” on Amazon: 1473 each of “Useful” (4 stars or more and 10 or more useful people), “Useless” (4 stars or more and 0 useful people), and “Low-rated” (2 stars or less) reviews. For the text data used in this experiment, we excluded in advance texts that were extremely short, such as those with only one or two words, and texts with excessively unnatural Japanese. For example, in the review text, the correctness of the content was not questioned because the purpose of this experiment was to check whether the text data could be interpreted as it is.

The experiment uses an LSTM model, which is an advanced version of an RNN, to improve the accuracy of the training. The training was done using LSTM, and the middle layer was one layer. The number of nodes in the middle layer was reduced to the extent that the classification accuracy did not fall below 95%. The learning rate was 0.1, the 11- and 12-norm coefficients were both 0.0001, and the number of trainings was 50.

C. Experimental Results and Discussion

First, the breakdown of the validity of the interpretations described by the participants (participant average) is shown in Figure 7. However, this breakdown was classified by one of the authors on the basis of the following definitions.

- Reasonable interpretation (reasonable): The correctness of the content can be confirmed from the source text and meets the “Purpose of interpretation”.
- Interpretation that cannot be judged as either valid or not valid (unknown): The intention of the content is not clear and cannot be judged as either valid or not valid.
- Unreasonable interpretation (unreasonable): The content of the interpretation is confirmed to be incorrect or does not meet the “Purpose of interpretation”.

This classification process was performed mechanically by the author based on the following procedure. In order to avoid

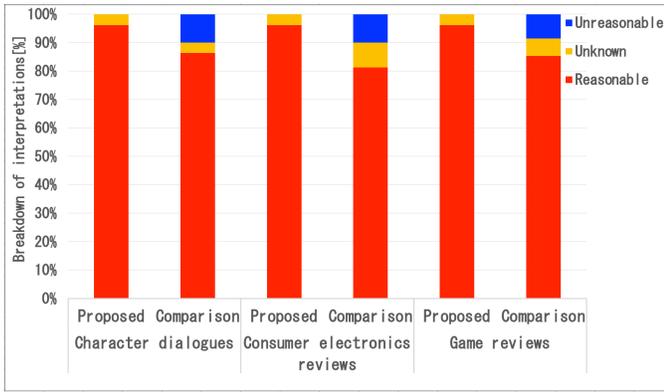


Fig. 7: Breakdown of validity of participant's interpretations (participant average)

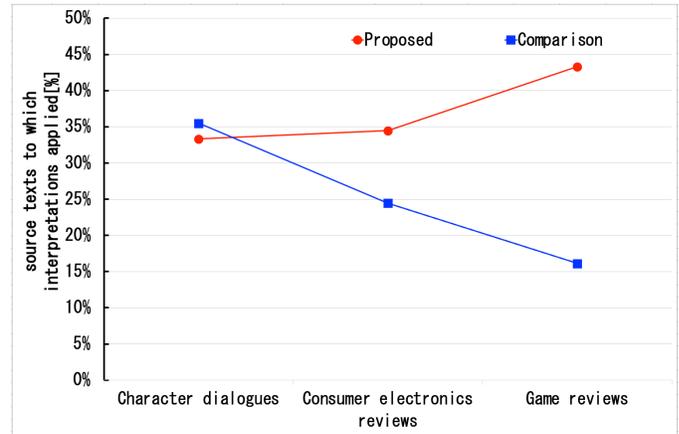


Fig. 9: Percentage of source texts to which participants' interpretations applied (participant average)

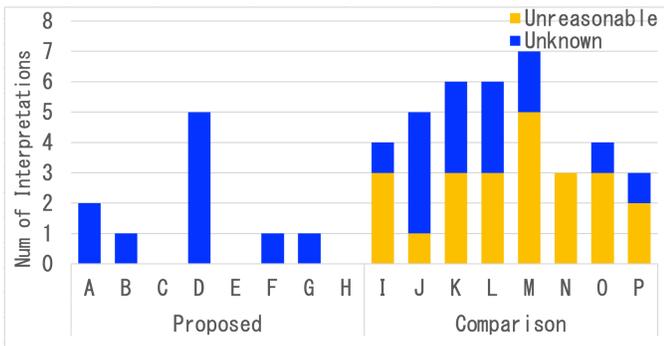


Fig. 8: Unknown and misinterpreted numbers by participant

any oversight, this process was repeated several times with a time interval between repetitions.

- 1) Check whether the interpretation matches the “Purpose of interpretation” in Table II. Interpretations that clearly do not meet the purpose are classified as “Unreasonable interpretations”.
- 2) The set of source texts (ORG) in which the features (words) of interest in deriving the interpretation appear is the target of the investigation. Interpretations for which the ORG does not exist are classified as “Unreasonable interpretations”.
- 3) If the ORG contains the content of the interpretation, it is classified as a “Reasonable interpretation”; if not, it is classified as an “Unreasonable interpretation”.
- 4) If the meaning of the interpretation is not understood, or if there are multiple possible meanings, and it is not clear whether the interpretation is included in the ORG in 3), classify it as an “Unknown interpretation”.

Table IV shows examples of interpretations that were actually classified as “Reasonable interpretation”, “Unreasonable interpretation,” and “Unknown interpretation” by the above classification procedure and the reasons.

Figure 7 shows that more than 97% of the interpretations with the proposed system were classified as valid interpreta-

tions, which confirms the correctness of the proposed system. The number of interpretations that were not valid was nearly 10% in the comparison system, but 0% in the results of the proposed system. Furthermore, unknown interpretations accounted for 5 to 10% in the comparison system, but less than 3% in the proposed system. This indicates that the proposed system has clearer intentions and more valid interpretations.

Figure 8 shows the number of “Unreasonable interpretations” and “Unknown interpretations” for each participant: A to H represent eight participants of the proposed-system group, and I to P represent eight participants of the comparison-system group.

Figure 8 shows that the number of participants who gave “Unknown interpretation” was 5 in the proposed-system group and 7 in the comparison-system group, and there was no significant difference between them. This indicates that all but one of the participants gave multiple “Unreasonable interpretations”. Therefore, it can be confirmed that the proposed system gave more valid interpretations regardless of individual differences.

Figure 9 shows the percentage of the source sentences that fit the interpretation given by the participants (participant average) . For each participant, the sum of the number of source texts that contain statements consistent with these interpretations is divided by the number of source texts per task (500 for the “Character dialogues” task, 1036 for the “consumer electronics review” task, and 1473 for the “game review” task), and the result is the percentage of source texts to which the interpretation applies. The results for the “Character dialogues” task were almost the same, but for the “Consumer electronics reviews” and “Game reviews” tasks, the proposed system was able to derive more interpretations that fit the source texts. For the “Game Reviews” task, the proposed system outperformed the comparison system by nearly 30%, indicating that the interpretation support network displayed with the proposed system was able to derive more typical interpretations that applied to a wider range of source texts.

Figure 10 shows the breakdown (participant average) of

TABLE IV: Examples of classification results and reasons for classification of participants' interpretations

classification result	example interpretation	reason for classification
Reasonable interpretation	Task 1 "Character dialogues": After the phrase "Don't get me wrong", the character says something that negates the previous conversation	"Don't get me wrong" is found in the source text, which negates the other person
	Task 2 "Consumer electronics review": Nozzle performance for carpets is considered to be a feature of the article	the description "about nozzles for carpets" was found in the source text
Unknown Interpretation	Task 1 "Character dialogues": The attribute "Tsundere" can be assumed to be strongly related to romantic relationships	it is difficult to determine which sentences in the source text are related to romantic relationships
	Task 1 "Character dialogues": Tend not to make negative comments	difficult to determine whether there are any "negative comments" in the text
Unreasonable interpretation	Task 2 "Consumer electronics reviews": Robot vacuum cleaners are not considered to be highly rated compared with other types of vacuum cleaners, etc.	there is no indication in the source text that robot vacuum cleaners are not highly rated
	Task 2 "Consumer electronics reviews": It is considered to be characterized by writing about product specifications and evaluations	does not achieve the purpose of interpretation as it applies to all the source texts

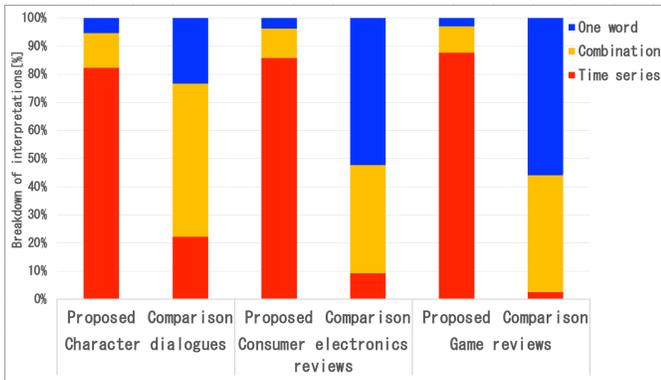


Fig. 10: Breakdown of features focused on by participants (participant average)

which features (one word, combinations, time series, etc.) the participants focused on for interpretation. However, the breakdown of the features focused on was classified by one of the authors on the basis of the following definitions.

- One word: A single interpretation is made from one word.
- Combination: A single interpretation is made from multiple words without considering the time-series relationship.
- Time series: A single interpretation is made from multiple words, taking the time-series relationship into account.

Figure 10 shows that more than 80% of the interpretations with the proposed system focused on time-series information of words. With the comparison system, only about 10% of the interpretations focused on time-series information between words, and the rest were word units and word combinations. This may be because it was easy to understand the time-series information of the words with the interpretation support network of the proposed system; thus, the participants could easily make interpretations focusing on the time-series of the words. With the comparison system, although the characteristic words of the top TFIDF were displayed, the connection between each word was unclear, and the participants often interpreted the words themselves or by combining words with similar meanings. This could be the reason why many of the participants in the comparison system led to wrong

interpretations that did not fit the source texts. Therefore, we can say that the proposed system performed a typical interpretation considering the time series of words.

Finally, Table V shows the purpose and examples of interpretations that were particularly common in the proposed system and the comparison system as a trend of interpretation for each task. The words in “[” and “]” indicate words that were actually displayed in the interpretation support network of the proposed system and in the word lists of the comparison system. In the examples of features of interest, the classification of whether the features of interest is time series or not is also indicated. In addition, since all interpretations answered by the participants were in Japanese, Table V includes both Japanese sentences and their translations.

Table V shows that many of the interpretations for the tasks “Consumer electronics reviews” and “Game reviews” consider the details of the product or game, such as what the review should focus on and descriptions of what people might be interested in, rather than the content of the product itself or the game, in the proposed system. Most of these interpretations focused on features of the time-series, suggesting that attention to the features of the time-series allows for interpretations that consider the text as a whole. Conversely, the comparison system resulted in many interpretations of the characteristics of the product itself and the content (genre) of the game, based on individual words. Therefore, in the comparison system, there were many interpretations that applied to only some products and games, and the percentage of source texts to which the interpretations applied was considered to have dropped.

On the other hand, in the task “Character dialogues”, many interpretations focused on patterns of time series in the proposed system and patterns of individual words in the comparison system, but in both cases, we confirmed a tendency for many descriptions of the unique expressions of the characters. This is because the average length of “Character dialogues” is only about 40 characters (about 20 words), and the full text is available in the original text display function whether the user selects a single word or multiple words, which may result in similar interpretations in both groups. The same reason can be considered for the result that the percentage of source texts to which the interpretation of the task “Character dialogues” applies was about the same in both

TABLE V: Trends in participant' interpretations (A: proposed, B: comparison)

Tasks	Purpose of Interpretation (number of items)	Examples of Noted Features	Examples of Interpretation
A: Character dialogues	Character-specific expressions (32 items)	Time series: “[ない] → [嫌い]” (“[not] → [dislike]”) etc.	“好きじゃないが嫌いでもない、相手への好意を示す際に曖昧な表現をする” (“She is ambiguous in expressing his fondness for the other person, saying, I don't like him, but I don't dislike him either.”) etc.
A: Consumer electronics reviews	Product accessories and other details (22 items)	Time series: “[付属]の後に[充電],[パック],[ノズル]という言葉が続いている” (“[attached] followed by the words [charging], [pack], [nozzle]”) etc.	“付属品についての詳しい情報が役立つ場合が多いと考えられる” (“I think more information about the attached accessories would be helpful in many instances.”) etc.
A: Game Reviews	Interesting Game Details (24 items)	Time series: “[史上],[オープン]と続いている” (“[ever] followed by [open]”) etc.	“史上最高と書くことで面白さが伝わりさらに最近人気のオープンワールドゲームという情報を入れることで興味を持たせられると考えられる” (“By writing that it's the best ever, we think it will convey the fun of the game, and by including the information that it's an open-world game, which is very popular these days, we think it will generate interest.”) etc.
B: Character dialogues	Character-specific expressions (34 items)	One word: “[ない]が上位に上がっている” (“[Not] is rising to the top of the list.”) etc.	“好きじゃないのように言葉を否定するのが特徴と考えられる” (“Like I don't like it, denying words is considered a characteristic.”) etc.
B: Consumer electronics reviews	About the product Itself (36 items)	One word: “[明るい]の単語が頻度が高い” (“[Brighter] words are more frequent.”) etc.	“ライトの明るさに関する記事が明確に書かれているものが多い傾向にある” (“Many of the articles tend to be clearly written regarding the brightness of the lights.”) etc.
B: Game Reviews	Genres users are looking for (21 items)	One word: “[ファンタジー]がジャンルとして出現している” (“[Fantasy] is appearing as a genre.”) etc.	“ファンタジー性をゲームに求めているユーザーが多いと考えられる” (“It is thought that many users are looking for fantasy nature in their games.”) etc.

groups.

In summary, we confirmed that the proposed system was able to derive typical and reasonable interpretations that were applicable to a wide range of source texts with a higher rate of correct answers than the comparison system. This can be attributed to the fact that the proposed system focuses on the time-series information between multiple words. We also confirmed that even in the case of short texts, such as in the “Character dialogues” task, the proposed system was able to derive typical interpretations at the same level as referring to words with high TFIDF values.

V. CONCLUSION

We proposed a classification-pattern interpretation support system to classify multiple text data with an RNN that can learn the time-series relationship of words and interpret the learned network. One of the features of the proposed system is that it can easily extract the time-series information of the learned features without learning on a special model by fitting the network structure of the learned recursive deep learning to an HMM. In the verification experiment, we confirmed that the proposed system can easily lead to a reasonable interpretation that covers a wide range of content of the source text from the classification patterns including the time-series information, even for users who are not familiar with deep learning.

In the future, we would like to change the input of the RNN to a distributed representation that includes information on the relationship between words, so that the interpretation can be more focused on the meaning of the words. We aim to build an interpretation environment for more complex deep learning networks, such as Bidirectional Encoder Representations from Transformers, by obtaining data from inside and outside the

training data to support the validity of the interpretation given by the user and presenting it to the user.

REFERENCES

- [1] M. Ando, Y. Kawahara, W. Sunayama, Y. Hatanaka, ‘Interpretation Support System for Classification Patterns Using HMM in Deep Learning with Texts’, In Proceedings of the Fourteenth International Conference on Advances in Computer-Human Interactions (ACHI 2021), pp. 64-70, 2021.
- [2] K. M. Tran, Y. Bisk, A. Vaswani, D. Marcu, and K. Knight, ‘Unsupervised neural hidden markov models’, In Proceedings of the Workshop on Structured Prediction for NLP, pp. 63-71, 2016.
- [3] H. Wang, K. Sreenivasan, S. Rajput, H. Vishwakarma, S. Agarwal, J. Sohn, K. Lee, and D. Papailiopoulos. ‘Attack of the Tails: Yes, You Really Can Backdoor Federated Learning’, 34th Conference on Neural Information Processing Systems (NeurIPS 2020), 2020.
- [4] D. Gunning, ‘Explainable artificial intelligence (xAI)’, Tech. rep., Defense Advanced Research Projects Agency (DARPA), 2017.
- [5] A. Fernandez, F. Herrera, O. Cordon, M. Jose del Jesus, F. Marcelloni, ‘Evolutionary fuzzy systems for explainable artificial intelligence: Why, when, what for, and where to?’, IEEE Computational Intelligence Magazine 14 (1), pp. 69-81, 2019.
- [6] J. Haspiel, N. Du, J. Meyerson, L. P. Robert Jr, D. Tilbury, X. J. Yang, A. K. Pradhan, ‘Explanations and expectations: Trust building in automated vehicles’, Companion of the ACM/IEEE International Conference on Human-Robot Interaction, ACM, pp. 119-120, 2018.
- [7] O. Goudet, D. Kalainathan, P. Caillou, I. Guyon, D. Lopez-Paz, M. Sebag, ‘Learning functional causal models with generative neural networks’, Explainable and Interpretable Models in Computer Vision and Machine Learning, Springer, pp. 39-80, 2018.
- [8] R. M. J. Byrne, ‘Counterfactuals in explainable artificial intelligence (XAI): Evidence from human reasoning’, Proceedings of the Twenty-Eighth International Joint Conference on Artificial Intelligence, IJCAI-19, pp. 6276-6282, 2019.
- [9] X. Yuan, P. He, Q. Zhu, X. Li, ‘Adversarial examples: Attacks and defenses for deep learning’, IEEE Transactions on Neural Networks and Learning Systems 30 (9), pp. 2805-2824, 2019.
- [10] G. Audemard, F. Koriche, P. Marquis, ‘On Tractable XAI Queries based on Compiled Representations’, KR Proceedings 2020 Special Session on KR and Machine Learning, pp. 838-849, 2020.

- [11] Q. Zhang, Y. Yang, H. Ma, Y. N. Wu, 'Interpreting CNNs via decision trees', IEEE Conference on Computer Vision and Pattern Recognition, pp. 6261-6270, 2019.
- [12] A. Barredo Arrieta, N. Díaz-Rodríguez, J. Del Ser, A. Bennetot, S. Tabik, A. Barbado, S. García, S. Gil-Lopez, D. Molina, R. Benjamins, R. Chatila, and F. Herrera, 'Explainable artificial intelligence (xai): Concepts, taxonomies, opportunities and challenges toward responsible ai', Information Fusion, vol. 58, pp. 82-115, 2020.
- [13] E. Tjoa and C. Guan, 'A Survey on Explainable Artificial Intelligence (XAI): Toward Medical XAI', IEEE Transactions on Neural Networks and Learning Systems 20 Oct. 2020, pp. 1-21, 2020.
- [14] J. Wagner, J. M. Kohler, T. Gindele, L. Hetzel, J. T. Wiedemer, S. Behnke, 'Interpretable and fine-grained visual explanations for convolutional neural networks', Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 9097-9107, 2019.
- [15] C. Panigutti, A. Perotti, D. Pedreschi, 'Doctor XAI: an ontology-based approach to black-box sequential data classification explanations', Proceedings of the 2020 Conference on Fairness Accountability and Transparency, 2020, pp. 629-639.
- [16] M Daniluk, T Rocktaschel, J Welbl, S Riedel, 'Frustratingly Short Attention Spans in Neural Language', ICLR, 2017.
- [17] Cookpad Inc., 'cookpad', <http://cookpad.com>, <link> 2022.06.01.
- [18] Amazon.com, Inc., 'Amazon.co.jp', <https://www.amazon.co.jp>, <link> 2022.06.01.

Utilizing Fuzzy Sets and Rule Engines for Intelligent Task Assignment in Industry 4.0 Production Processes

Gregor Grambow, Daniel Hieber and Roy Oberhauser

Dept. of Computer Science

Aalen University

Aalen, Germany

e-mail: {gregor.grambow, daniel.hieber, roy.oberhauser}@hs-aalen.de

Abstract—Today’s Industry 4.0 Smart Factories involve complicated and highly automated processes. Nevertheless, certain crucial activities such as machine maintenance remain that require human involvement. For such activities, many factors have to be taken into account, like worker safety or worker qualification. This adds to the complexity of selection and assignment of optimal human resources to the processes and overall coordination. Contemporary Business Process Management (BPM) Systems only provide limited facilities regarding activity resource assignment. To overcome these, this contribution proposes a BPM-integrated approach that applies fuzzy sets and rule processing for activity assignment. Our findings suggest that our approach has the potential for improved work distribution and cost savings for Industry 4.0 production processes. Furthermore, the scalability of the approach provides efficient performance even with a large number of concurrent activity assignment requests and can be applied to complex production scenarios with minimal effort.

Keywords—Business Process Management Systems; Fuzzy Logic; Rule Engines; Resource Allocation Algorithms; Assignment Automation.

I. INTRODUCTION

With this paper we extend our previous work [1], where we first introduced our intelligent assignment concept. We expound on the concrete workflow and implementation of the utilized fuzzy sets for task assignment in Industry 4.0 processes. Furthermore, we present a functional implementation of the Rule Interface, introduced in our previous work as a theoretical concept. Finally, we tie together our other work, successfully implementing the intelligent task assignment.

“Industry 4.0” stands for the fourth industrial revolution driven by digitalization [2]. Highly automated Smart Factories enable more efficient and individual production methods as well as greater customer focus. This includes the comprehensive control and organization of the entire production value chain by utilizing real-time data processing across all production stages. Cyber-Physical Systems (CPS) [3], which consist of information technology (IT), machines, and built-in sensors, form a unit that enables comprehensive optimization of production with regard to criteria such as costs, resource consumption, quality, or availability. While a strong focus on autonomous systems and the highest possible degree of automation exists, in highly complex processes human involvement remains indispensable. Often the production process

depends on activities, in which people intervene, perform complex activities and make important decisions.

Such higher-level business and production processes are typically governed by Business Process Management Systems (BPMS) [4], also known as Process-Aware Information Systems (PAIS). BPMS are in charge of the sequencing of the different activities belonging to a business process including automated activities and those processed by human agents. The success of any BPM process realization can be endangered by excessive activity automation and poor design of work assignment strategies [5]. Therefore, assigning the optimal agent to an activity and vice-versa is a time-consuming but necessary task with every BPMS. In most BPMS, so-called Staff Assignment Rules (SARs) (or resource allocation) are utilized to achieve this. Yet this area has not received sufficient research attention, as indicated by the survey by Arias et al. [6].

Moreover, in Industry 4.0 production scenarios, many different factors have to be taken into account to select an agent that can process an activity in an efficient and effective manner. An obvious example for such factors is the qualification of the agent, who must have the necessary skills and abilities to correctly execute the activity without being overqualified (and thus incurring unnecessary cost overhead). Usually agents with a much higher qualification level should not be assigned to a particular activity. Such optimizations should also consider balancing the agent workload to not overburden an agent while others are idle.

In large production facilities, the physical location of the agents and where the activities are to be performed also play an important role. An example are maintenance activities that have to be executed from time to time across a large number of production machines at a large facility. If not optimized properly, agents may waste a substantial amount of time in transit to activities, analogous to the well-known Traveling Salesman Problem [7]. Due to the high complexity of Smart Factories and their CPS, involving specialized external (maintenance) workers with specific knowledge to maintain a system can incur additional costs. To contain these costs, utilization of internal employees should be preferred if possible, depending on the urgency, availability, and qualification levels. In modern production, worker safety is also an important factor that is usually regulated by respective laws, which address hazards

such as chemical, electrical, heat, and noise and may not be adequately tracked by automation systems.

When taking such factors into account, it becomes evident that standard BPMS SARs are insufficient because they are only capable of determining if an agent is available to perform an activity, but cannot readily determine the degree of suitability. Fuzzy logic's [8] fine granular classification between 0 and 1 provides a way to overcome the limitations of simple Boolean logic and determine a specific assignment score for each agent for each possible assignment. Automating such a generic and recurring activity can optimize work efficiency and manpower cost, while reducing employee frustration when automated systems seem inflexible or make unsuitable assignments. By combining fuzzy logic with rule processing, pre-filtering allows the fuzzy-specific areas to be factored from the rest of the more obvious rule-based resource allocation problem, utilizing the best of both fuzzy logic and rule engines.

In prior work [9][10][11], we also developed a different approach for contextual process management that did not rely on Fuzzy Sets but rather utilized an adaptive process management engine. software engineering processes and did not use fuzzy sets or involve the complex specifics of Industry 4.0 nor processes with integrated AR support. The main focus was extending processes with properties to enable automated software quality assurance and support collaboration of software engineers. This was realized via automatic process adaptations.

In this paper, we contribute an approach for activity assignment in Industry 4.0 projects that takes the aforementioned factors into account. By applying fuzzy sets and a rule engine, fine-grained levels of suitability are integrated to improve resource assignment results. To demonstrate its feasibility, we integrated our solution with a common BPMS. We further extended our previous version of the Intelligent Assignment Component with the integration of a rule engine, which was only theoretically discussed in our earlier work.

The remainder of this paper is structured as follows: Section II highlights related research and background information. Section III then describes the general concept and an initial solution approach, while Section IV details the concept for our Intelligent Assignment Component (IAC). In Section V, we provide specific implementation details focusing on the IAC while addressing the overall prototype. Then in Section VI we evaluate our solution. Finally, Section VII provides a conclusion and outlook on future upcoming work.

II. RELATED WORK

In literature there are numerous approaches for activity assignment optimization utilizing different algorithms like fuzzy sets. Kubler et al. [12] provide a survey of the application of Fuzzy Logic in combination with Multiple Criteria Decision-Making, and within the category resource allocation, Shahhosseini and Sebt [13] is the only example of its application to human resource allocation. However, it is specific to construction companies, centers around four specific human roles and lacks an integration strategy with BPMS. Similarly, Kłosowski

et al. [14] also discuss a fuzzy model for assigning workers to production activities. The main focus of their approach is employee assessment and a rich set of properties. However, for our use case, the model is too generic and contains unnecessary properties, while at the same time neglecting other important factors like worker safety or location. Furthermore, it also lacks BPMS integration concepts. Seifi et al. [15] apply fuzzy logic to optimize human resource allocation for project planning in small-to-medium sized organizations and does not consider live processes with a BPMS. In contrast, our work focuses on the Industry 4.0 production and is integrated with a BPMS.

Kluza and Nalepa [16] provide a formalized model combining a procedural business process model with Attribute Relation Diagrams for rules, and describe an algorithm that can generate an executable BPMN model with decision table schemas for rules in XTT2 representation. In contrast, our approach does not address automatic generation of models, and while supporting the simplicity of rules, our approach can address more complex problems with resource allocation by using fuzzy logic.

Antonelli and Bruno [17] deal with an Industry 4.0 topic: activity assignment in human robot collaboration. This approach splits the activity assignment problem into activity classification with a decision tree classifier and activity assignment with a decision-making algorithm. However, the approach does not address BPMS integration and relies on Boolean rather than fuzzy values, which makes it somewhat synthetic. In addition, worker safety is not taken into account.

Another approach for activity-resource assignment that applies fuzzy logic is presented by Xu et al. [18]. It contains a comprehensive but complicated fuzzy model targeted at collaborative logistics networks comprising logistics service integrators, activity contractors, and resource providers. Thus, the model cannot be used for the assignment of single workers in Industry 4.0 production. Finally, a category of approaches similar to Simpson and Roberts [19] utilize various algorithms like Bayesian methods, heuristic algorithms or game theoretic approaches for activity assignment in spatial crowdsourcing. As this domain has rather specific properties on which the algorithms rely, they also cannot readily be applied to Industry 4.0 production and for similar reasons, BPMS integration is not included in these approaches.

Approaches using rule-based resource allocation include iDispatcher [20], which focuses on the insurance domain using ILOG JRules. It does not address non-human allocations, context-awareness, or fuzzy problems. Li et al. [21] use the Drools rule engine and focus on human resource allocation for the IT service order rather than the Industrial Internet of Things (IIOT) domain. While mentioning the term workflow engine, it does not state which one nor explicitly address how it is practically integrated with a BPMS to do the assignment for an activity. Havur et al. [22] use Answer Set Programming with the clasp solver on timed Petri Nets using RDF Schema as a resource ontology for the railway engineering domain. In contrast, our work shows integration with commercial BPMS

and utilize the combination of a rule engine and fuzzy logic. Sikal et al. [23] utilizes process mining for resource variability discovery, but does not explicitly address rule modeling, fuzzy problems, nor the integration in a specific BPMS, and presumes this data is available a priori for analysis. Erasmus et al. [24] apply the Fleishman taxonomy for specifying activities, human resources, and their ability-based allocation during runtime for a manufacturing case study. The information from their method is stored in an SQL database and Java methods are used to make the assignments with the Camunda BPM. It does not address non-human allocations nor fuzzy problems. Vasilecas et al. [25] describe a rule- and context-based approach of dynamic business process modeling and simulation. It consists of a custom .NET-based implementation without BPMN compliance, is focused on simulation, and considers an ordering process.

Further non-fuzzy resource-allocation related work includes Ihde et al. [26] who describe a resource-aware extension framework to a traditional BPMS for process designers to specify resource allocation constraints, enabling external allocation services with different algorithms to process activities. They did not apply fuzzy logic nor did it involve a Industry 4.0 or production situation. Tan et al. [27] propose an optimal resource allocation strategy for cooperative task scheduling in cross-organizational business processes. It focuses on team formation, considering professional and cooperative ability, analyzing process event logs for an insurance claim business process. In contrast, our work extends an actual BPMS, uses fuzzy logic, and is applied in the Industry 4.0 domain.

III. SOLUTION APPROACH

While different fuzzy-based approaches for activity assignment exist, they are often rather generic and complicated, or too specific and tailored to a certain domain. Moreover, they typically do not address integration with contemporary BPMS. To overcome these limitations and be able to create a usable system for Industry 4.0 scenarios, we focus on a more concrete model and a specific component executing the activity assignments while addressing integration with current BPMS.

To achieve suitable assignments in a practical and applicable manner, our approach addresses these requirements:

- 1) The system shall calculate an assignment score that reflects the suitability level of agents for handling a specific activity.
- 2) The runtime shall be capable of handling a large number of concurrent assignment scoring requests efficiently.
- 3) Integration into BPMS shall be readily feasible.

To maximize the efficiency of optimization options and support easy integration into various BPMS, a new system for handling assignments is created. By decoupling the assignment process from the BPMS, a separate component can be implemented solely for the assignment process, permitting better performance optimization without the constraints imposed if one were to internally extend a specific BPMS. Furthermore, this decoupling via a generic API supports a generic approach

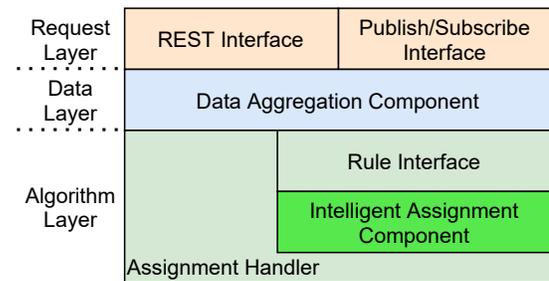


Figure 1. Assignment and context engine conceptual architecture.

that can support integration across a much wider range of BPMS. The conceptual architecture of the novel Assignment and Context Engine (ACE) providing such functionality can be seen in Figure 1.

The ACE uses a layer pattern, which is further subdivided into components, with each layer contributing to the final solution. Via the modular layers, if desired, the Data and Algorithm Layer could be directly integrated into BPMS (potentially enhancing performance). Alternatively, only the Assignment Handler or its individual components could be directly integrated in a BPMS (with a reduced set of features). Thus, we hitherto focus on the ACE as a holistic solution to fulfill the aforementioned requirements.

The public REST and Publish/Subscribe (Pub/Sub) Interfaces in the Request Layer are used as a BPMS- and programming-language-independent interface, allowing the usage of the ACE with any BPMS supporting BPMN 2.0 or later. The REST and Pub/Sub interface can be used interchangeably as required by the concrete use-case, depending on the message volume and other factors. This standard offers a wide range of elements to integrate external services and functions [28] [29]. The integration is based on a dual activity concept. A utility-activity requests an assignment for the execution-activity succeeding it in the process workflow. For the utility-activity, two approaches in BPMN 2.0 are possible and should be chosen according to the capabilities of the applicable BPMS (Figure 2).

The synchronous variant utilizes a Service Task to request the assignment from the ACE synchronously. The service activity receives the required data from the process and then awaits the calculated assignment. Finally, it assigns the agent with the highest suitability level to the activity. The asynchronous variant utilizes a Script Task that obtains all the required values itself, accessing the BPMS and then requesting an assignment asynchronously. The BPMS can then ignore the process until the assignment is calculated and no resources have to await a response. As soon as the ACE finishes the calculation, it calls the BPMS API and assigns the best fitting agent to the activity itself. With this approach every state-of-the-art BPMS can easily be integrated (requirement 3) and one ACE could even support multiple BPMS at the same time.

Once a request is received by the backend, it is validated by the Data Aggregation Component (DAC). If all required information is present, the request is passed on to the Assignment

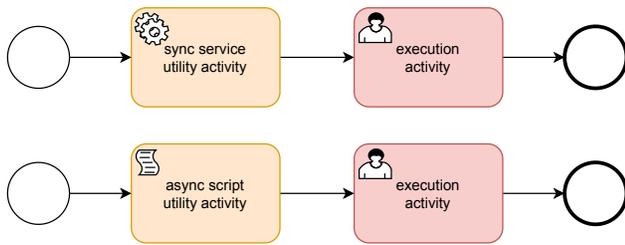


Figure 2. Activity solution variants (synchronous variant on top, asynchronous variant on bottom).

Handler. If some data is still missing the DAC can receive this via predefined external sources, e.g., a database containing the agents attributes (like position, or qualification) or directly via the BPMS API if endpoints to request additional information are provided.

The Assignment Handler is responsible for the overall assignment process, this is split into three parts:

- Checking preconditions
- For the requesting activity, determining the assignment score for each agent; and
- Triggering the assignment process in the BPMS

To reduce the overall load on the IAC, and to enable task filtering on external values (like sensor measurements or available inventory), rule-based methods are employed. By creating a Rule Interface as a facade, external rule engines can be readily connected to the ACE. These can utilize efficient algorithms to validate if preconditions are fulfilled or if something prevents the successful assignment and execution of the task. This rule engine integration can also be used to pre-filter agents, removing incapable agents from the actual assignment calculation. This can be advantageous when involving a high number of agents per task, as chaining algorithms are generally more efficient than fuzzy sets (requirement 2).

In order to provide the desired assignment score with a fine granular suitability level, fuzzy sets are chosen. As seen in Section II fuzzy approaches are able to generate very precise assignment scores (requirement 1) in an efficient way (requirement 2). This is an improvement over currently employed chaining-based SARs, which are capable of calculating accurate assignments, but lack the capability to differentiate between suitable agents, and thus do not provide overall optimal assignments. While a Machine Learning (ML) approach would also be feasible, the fuzzy sets provide some striking advantages. For fuzzy sets, no preexisting datasets are required, and necessary weights can be (re-)configured according to empirical manual feedback or settings rather than requiring actual digitalized data for analysis and training. This enables more traditional companies with weak digitalization and low to no sensor coverage an intelligent assignment capability without a costly and long running preparation phase. Also, this capability can transfer the intelligent assignment with adapted weights instantly to all parts of its production and workflow. Moreover, our intelligent assignment approach avoids the (costly) training phase typically required by ML

approaches.

While it may seem possible to achieve similar functionality with an alternative non-fuzzy approach, due to the benefits and arguments noted above, we believe this would require far more work to achieve the equivalent out-of-the-box functionality, flexibility, and maintainability that a fuzzy set approach provides (without necessitating preexisting data or training).

For determining the assignment score and assigning the most suitable worker, the ACE can either directly assign the agent via a REST-API (present in many of the most popular BPMS), or the assignment could be conducted in the sync service utility activity via script access to the BPMS from within the process itself.

Due to the complete decoupling of BPM and ACE, the latter can be scaled independently of the scaling of the BPMS, such that a high workload on one of these engines does not affect the performance of the other. The separation further allows the implementation of an optimized multi-processing and scaling functionality, guaranteeing optimal efficiency even at high load (requirement 2). The performance optimization takes place at different levels. First, a multi-threading approach is utilized in the Request Layer following reference architectures for REST and Pub/Sub APIs. The subsequent handling of the request in the Data and Algorithm Layer is handled in a separate process decoupled from the Request Layer. To further accelerate large assignment calculations, the IAC has its own scaling function introduced in Section IV.

Figure 3 shows simplified workflow graphs for two BPMN processes using the IAC to assign an agent to its execution activity. In Figure 3A) this is done via the concept described above, utilizing an asynchronous script utility activity. This activity requests an assignment from the ACE via REST and instantly receives an empty response to complete the utility activity. Afterwards, an external rule engine is called via REST using the Rule Interface. Receiving the response, the Assignment Handler then triggers the IAC to calculate the assignment scores via fuzzy sets. Afterwards, the Assignment Handler assigns the execution activity via the BPMS REST-API to the most suitable agent. The execution activity is available all this time (right after async script utility activity is completed by the empty REST response from the Assignment Handler), however, it is not assigned to any agent until the Assignment Handler does so via the REST API.

In Figure 3B. the assignment is handled via a synchronous service utility activity, calling the rule engine directly via a REST call and incorporating the IAC directly in the script activity itself. This approach supports the execution activity during the assignment process and does not allow refined scaling options outside of the BPMS itself.

As previously mentioned, it is also possible to call the Assignment Handler via a synchronous service utility task, merging the two approaches displayed in Figure 3. This works similarly to asynchronous script task approach, with the difference being that the assignment is handled by the service task rather than via an API call from the Assignment Handler.

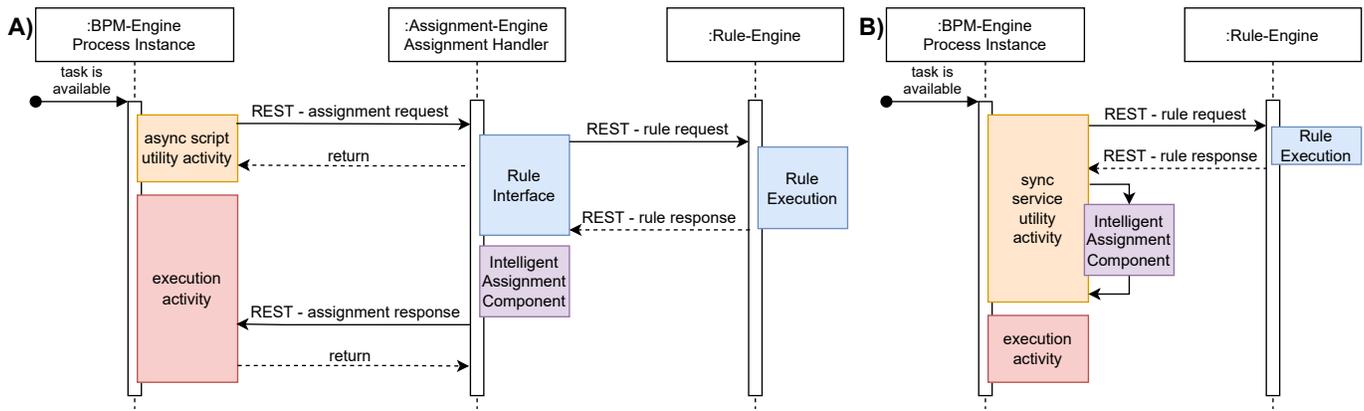


Figure 3. Different integration options in a BPM-engine with the utilization of a rule engine.

IV. INTELLIGENT ASSIGNMENT COMPONENT

The IAC is a standalone component of the ACE Assignment Handler. Containing the fuzzy logic for the assignment calculation, it is the functional core of the engine. This section highlights the conceptual decisions behind the component and details its internal structure.

A. Models

In order to compute meaningful assignment scores, the IAC requires a custom set of models. This is provided by the Assessment Criteria which are supplied as part of each activity and agent data set. These data sets can either be directly sent by the BPMS, when a new assignment request is sent, or be actively collected by the DAC via the BPMS API and connected data sources. For the second version the ID of all available agents and the activity must still be provided as part of the assignment request.

These models were also used as a foundation to create the Context and Augmented Reality eXtension (CARX) for BPMS [30]. This extension includes a modeler to create new BPM process templates, which contain all information required to enable an assignment via the IAC, while being fully BPMN 2.0 compliant and supporting the easy addition of information to new processes or the upgrade of existing processes.

1) *Assessment Criteria Model:* The Assessment Criteria consist of five parameters (oriented on real-world examples) that define the values for determining the assignment. These are required in activities and agents used with the IAC. It can be viewed as an interface required by all data and components connected to the assignment.

Distance: calculates the distance between agent and activity position optimizing assignments regarding the travel distance. Position objects contain 3D coordinates with numeric values for X, Y and Z.

Qualification: calculates the difference between the required qualification for an activity and the existing qualification of an agent. It answers the Boolean question if the agent is capable of performing the activity, and permits the determination of a possible over-qualification to prevent utilizing expensive agents on trivial activities. Qualification objects consist of the

four parameters: "electrical", "computer", "engineering" and "bio_chemical", which represent the different skills of agents or activity requirements in this area. As this skill cannot be calculated automatically and must be defined by humans, each parameter will be represented by a number between 0-10. This provides an accustomed scale to rank skill and requirements instead of a default fuzzy scale from 0-1, which is more abstract and an unusual scale for people.

Hourly Rate: calculates the extra cost of using a given agent for an activity per hour in Cents. This prevents the usage of external/temporary workers that incur extra costs if a similar qualified employee is available. This should not include the salary of permanent staff, as their salary is independent of their utilization rate. The cost is represented by an integer to prevent floating errors.

Workload: calculates the capacity utilization of agents, preferring agents with few enqueued activities and preventing overloaded agents from enqueueing additional activities. Thus, load balancing between resources and compliance with labor protection regulations can be supported within the algorithm. This could either be added to the agent itself by the BPMS, or can be calculated using the agents work list, which is present in all BPMS. The parameter is represented by an integer value.

Danger Level: calculates if an agent can safely perform an activity. As some activities have special hazards and given safety regulations, only agents with an appropriate safety clearance can be assigned to these. The Danger is thereby defined by an object consisting of the four parameters: "noise", "heat", "electrical", and "chemical". The separate values are represented by float values between 0 and 1. This provides an abstract concept, but can easily be modified for more concrete parameters as required for a given concrete use case.

2) *Activity Model:* This model can consist of any BPM activity extended with the Assessment Criteria except Hourly Rate and Workload, since the cost of an activity is irrelevant for its optimal assignment, and an activity itself has no workload in the context used by the IAC.

As the model extends BPM activities the activities priority remains intact and is respected by the IAC during activity assignment.

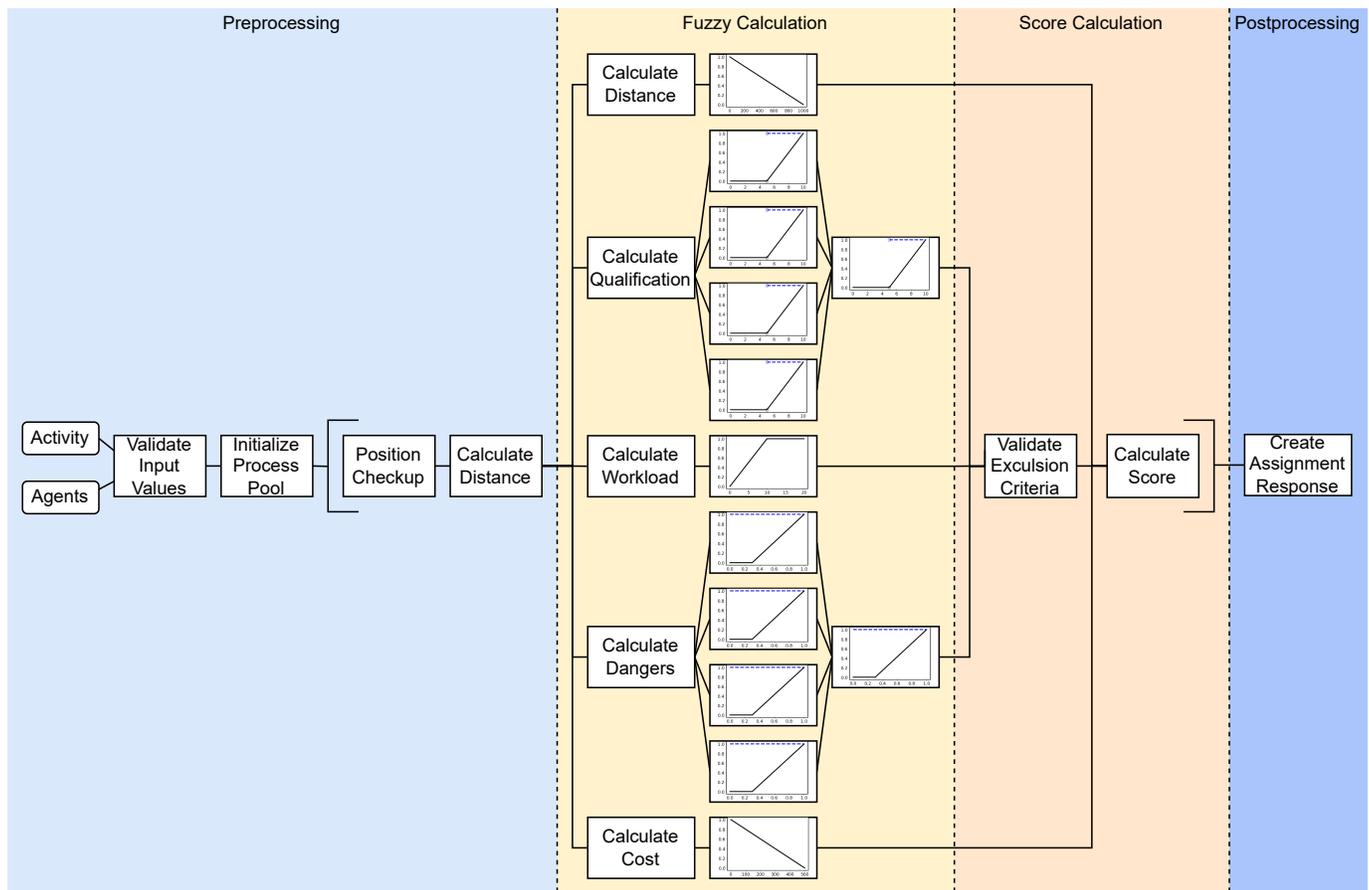


Figure 4. IAC - workflow tree.

3) *Agent Model*: The model consists of a BPMS human resource (user) extended with the Assignment Criteria as attributes. In a minimal engine configuration, such a resource might only contain an ID. In contrast to the activity, all criteria are mandatory, as they all provide valuable data for calculating optimal suitability levels. The Danger Level object is renamed to Danger Threshold on the agent level for a more descriptive and easier-to-understand naming. If a task is assigned to an agent the danger level of the activity is subtracted from the agent's threshold, preventing an overload with too many dangerous activities, which could lead to potential labor law violations and increased incident risk. After a resting period, the agent's danger levels are reset (e.g., at the start of the agent's next shift or after a longer break).

B. Overall Assignment Algorithm

The internal algorithms in the IAC are based on a Fuzzy Logic approach. In contrast to ML, no existing datasets are required, only a scheme of the data is mandatory to configure the fuzzy sets. However, the same level of fine calculation of the suitability score is possible, as opposed to the simple calculation of suitability based on chaining. As described earlier, the activity and agent list are provided to the component either directly by the BPMS or via the DAC according to the

forementioned models, and can therefore be directly supplied to the algorithms as parameters without further aggregations or parsing. After executing the algorithms, the IAC will return the suitability level for all provided agents to the Assignment Handler.

To speed up the processing time for large numbers of agents, the IAC can run calculations in a multi-processing configuration with multiple available modes. This allows an optimal resource allocation concerning the concrete assignments, rather than a general solution that could slow simple assignment calculations or non-optimally benefit more complex calculations.

An approximate workflow of the IAC can be seen in Figure 4. However, while parts of the fuzzy calculation are shown in parallel, it is actually executed sequentially. This format was chosen for its space-efficient layout. The order of the different calculations does not matter for the final result. The block displayed in brackets can either be executed in parallel via multi-processing or sequentially as a loop. The figure also contains images of the used fuzzy models to provide a general idea how they look. Figure 5 displays the over-qualification model in detail.

1) *Preprocessing*: After providing the activity and agent data to the IAC, some preprocessing steps take place (c.f.

Figure 4 Preprocessing). While the input values are already validated and per definition complete, the position entry for tasks is not necessarily set, as other defaults can be used. Namely, the position of a connected machine - if no positional constraints are enforced, and the position of the agent. Therefore, if no position is set, "Validate Input Values" attempts to determine the position of the machine connected to the task. If no machine is set, the value is left empty.

For the second step "Initialize Process Pool", the configuration is checked for the multi-processing flag. If this flag is set, it is checked if a pool amount is set. If no pool amount is set, the number of processes in the pool is set to $n - 2$ where n is the number of cores of the host system. A process pool is created and the following steps shown in brackets are executed in parallel by the different processes for each agent supplied to the IAC. Else, if no multi-processing flag is set, they are executed in a loop, once for each agent, in a sequential manner.

"Position Checkup" is the final agent-specific validation of the position. If, in earlier steps, no position was assigned to the activity, it is set to the agent's current location. This is required, as the fuzzy set for distance requires a position value of the activity, even if the activity itself is not location-dependent. If the activity position is already initialized (e.g., via the activity context or a connected machine) this step is skipped.

The final preprocessing step "Calculate Distance", calculates the Euclidean distance between the agent and activity. This is done by transforming the position values of agent and activity into three dimensional vectors (x, y, z) and deducting them. This can be used as a simple orientation. In a finalized productive system, however, this should be replaced by a more refined path calculation algorithm, providing a more resilient calculation for the following fuzzy set.

2) *Fuzzy Calculation*: After the preprocessing steps, the calculation of the actual fuzzy models can begin. Overall, the agent with the highest score is preferred. Therefore, all Assignment Criteria for each requested agent are calculated distinctly, with 1 being the best score possible and 0 the worst. Besides the score calculation, the fuzzy sets also contain certain exclusion criteria. In this section we only mention when they are reached, with a brief description why they are chosen. In the next section, we further elaborate these exclusion criteria in detail. The concrete calculation for each value is conducted as follows:

Distance: this algorithm uses the distance calculated in the corresponding preprocessing step as a basis. Distances between 0 and 1000 are mapped to a fuzzy value from 1 to 0, where all distances above 1000 are also mapped to 0, in order to simplify the handling of large distance values. Therefore, agents in close proximity to the activity are preferred in the assignment, and all agents further than 1000 units away are heavily discriminated by the fuzzy set. However, as this is no exclusion criteria, even agents further than 1000 units away could still be assigned to the activity if they are the overall most suitable agent.

Workload: the workload can take values between 0 and 20 at

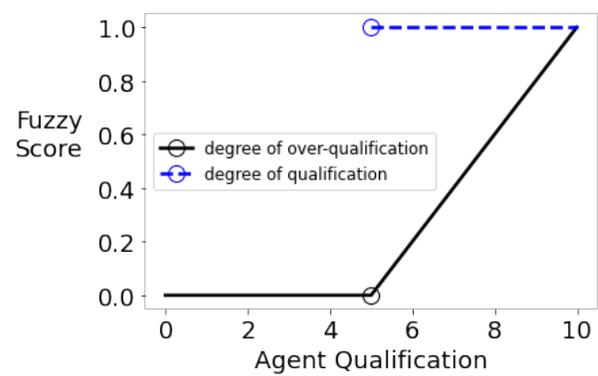


Figure 5. Qualification fuzzy model showing fuzzy score of agent over-/qualification for a task with qualification requirement 5.

maximum. Values between 0 and 20 are mapped to the fuzzy value between 1 and 0, while values equal to or greater than 10 are mapped to 0. This still allows the assignment of new activities to workers that already have 10 or more activities assigned to them, but prefers those with smaller workloads. Further, if the workload has a value of 20, the score is set to 0 and triggers the overloaded exclusion criterion, preventing the algorithms from assigning any more activities to this specific worker, thus preempting the overburdening of agents.

Danger Thresholds: for each danger value, a separate fuzzy set is calculated. After the disjunct calculations, all values are added to a common fuzzy domain and weighted according to a configuration (cf. Figure 4 "Calculate Dangers", first each domain is processed isolated, then all values are combined in a single domain). In the default case, all values are weighted the same, leading to a 25% weight per value. All danger values between 0 and 1 are mapped to fuzzy values from 0 to 1, where all values below the activity's danger level are 0 and trigger an exclusion criterion. This addresses labor law regulations while increasing worker safety. All values above the requirement through the maximum danger threshold of 1 are mapped between 0 and 1. An agent who approximately meets the requirements can therefore work on the activity but gets a score of 0. This prefers agents with higher danger thresholds, as they are most likely more experienced and more rested than agents with lower danger thresholds. While a valid agent who barely fulfills the requirements and an agent who fail the requirements both get a score of 0, a final filter in the "Validate Exclusion Criteria" Algorithm of the Score Calculation block removes all invalid agents from the assignment.

Qualification: the qualification is calculated in three separate fuzzy models. First, the four values of qualification (electrical, engineering, computer, bio_chemical) between 0 and 10 are compared to the values of required qualification of the activity between 0 and 10 via separate fuzzy sets similar to the danger levels. All values below the activity's requirements are assigned to 0 and trigger an exclusion criterion, as the agent is technically not capable of performing the requested activity. All values above the requirement are assigned to 1. Subsequently, the degree of over-qualification is calculated in the

TABLE I. IAC EXCLUSION CRITERIA.

Criteria	Condition	Explanatory note
Qualification	Agent qualification below activity requirement	This prevents agents from being assigned to activities they are formally not capable of, potentially reducing rate of errors, incident risk and execution time.
Workload	Workload of agent greater or equal to 20	This prevents the overburdening of agents with too many activities. Also, the general time before an activity is executed is reduced, as they are split more equally between the available workers.
Danger level	Agent's danger threshold below activity's danger level	This prevents potential labor law violations and increases the safety of activity executions.

second fuzzy model. Starting from the required qualification up to the max qualification of 10, each qualification value is assigned to a fuzzy value between 0 to 1, where 0 perfectly fits the required qualification value and 1 is the maximum amount of over-qualification possible. Afterwards the over-qualification is subtracted from the qualification value resulting in a value between 0 and 1 called degree of qualification, where 1 is a perfect fit without over qualification and 0 is a maximum over-qualification.

These values were all empirically selected and provided good results during our early tests. In a productive environment they could be applied for a base configuration. However, we recommend the selection of own values tailored to the concrete environment.

Figure 5 displays the two fuzzy models described above for an activity's qualification requirement of 5. The agent's score value for the (under-)qualification is displayed as the blue dashed line, where the fuzzy set for all possible agent qualifications below 5 returns 0, and jumps to 1 for all values fulfilling the requirements. The over-qualification is highlighted as a solid black line, the transition points between the qualification score of 0 to 1 are marked by circles. All agent qualifications below and at the required activity qualification of 5 return 0, representing no over-qualification. Above 5 the agent's qualifications are slowly assigned to a fuzzy value between 0 and 1, representing an increasing degree of over-qualification.

After this, two steps are conducted for all four properties of an agent's qualification. The four separate degrees of qualification are added to the final fuzzy domain and weighted according to the configuration. The process is identical to the Danger Threshold calculation and also uses a 25% weight distribution per value as a default. The resulting value is used as the qualification in the final calculation of the score, preferring qualified agents with as low an over-qualification as possible.

Hourly Rate: the hourly rate is mapped to the fuzzy value from 1 to 0 for values from 0 to 50000 (being equal to 500€ following the integer data format). All values over 50000 are set to 0. This prefers agents with low additional cost (like employees) over external workers, costing extra money and therefore improves the economic efficiency of the process.

3) *Score Calculation:* In the score calculation step, two algorithms are employed. First, the tree aforementioned exclusion criteria are involved. These could be triggered by the calculation of the fuzzy sets as described above. Table I lists them in detail.

To check if an exclusion criterion was triggered, the fuzzy value for each of the three domains is checked. If one of these values is 0, it is further checked if the agent's value is below the task's value. If this is also the case, the exclusion is triggered. Following this, the assignment score is set to 0 and the Score Calculation step is completed. As a fuzzy value of 0 is per definition valid (e.g., just fulfilling the qualification requirement also receives a fuzzy value of 0), comparing the agents value against the tasks value is required. Setting the fuzzy value to another value, e.g., -1, is not possible, as this would interfere with the calculation of the combinatorial fuzzy sets for qualification and danger.

Another approach could be throwing the exclusion criteria as an exception right after encountering it. This could potentially reduce the computing time of the IAC, as no further calculations would be required after encountering the first exclusion criteria. While this coding-by-exception approach is generally considered bad practice, it would be possible to move the "Validate Exclusion Criteria"-block right after each fuzzy model containing exclusion criteria, rather than having one block after all calculations.

After the exclusion criteria are checked and none was thrown, the agent's assignment score for the activity is calculated. This is once again achieved by a fuzzy set, combining all scores from the five previous fuzzy sets and once again applying a weight to them, similar to the calculation of the danger and qualification values. The weights for this final score calculation can either be supplied on a per assignment request basis directly by the BPM process, or via a default value in a configuration file. The first approach is quite beneficial if an activity is very one-sided and strongly focuses on one assignment criteria, e.g., a focus on distance alone if a fire has to be extinguished.

When the final fuzzy value has been calculated, it is multiplied by 100, giving the agent a final assignment score between 0 and 100 for the activity, while 0 means the agent is completely unsuited and 100 is a perfect fit.

4) *Postprocessing:* As the last component of the IAC, the "Create Assignment Response" algorithm is triggered. This takes all calculated assignment scores together with the agent's id as an identifier and orders them by score, putting the highest score on top. Afterwards the final assignment score structure is returned to the ACE for further processing.

V. IMPLEMENTATION

The ACE has been implemented as part of the Augmented Reality Process Framework (ARPF) [31], additionally incor-

porating AR and context support for workers during the execution of tasks. However, following the modular solution concept, it is still possible to use the IAC as a standalone module (e.g., directly integrated in a BPMS) or to use the ACE as a standalone product without the ARPF. CARX also utilizes ARPF (and therefore the IAC) as the CARX BPMS IIoT Extension Framework.

Like ARPF, our prototype of the ACE with focus on the IAC is implemented using Python. This approach was chosen for its fast prototyping capabilities while still providing performant libraries and refined multi-processing logic. As a base image for the ACE, a Django server was created, providing the most powerful REST-Server available for Python. In contrast to other Python server-frameworks, Django offers not only fast and simple prototyping capabilities, but can also be scaled up to a performant production deployment. To provide the required REST interface, the Django REST framework was integrated. A Pub/Sub interface was implemented using Python paho, the Python MQTT [32] framework from Eclipse. Architecturally the DAC is adjacent to the REST Layer. It can invoke REST requests on its own, aggregating all required data from the BPMS or configured external data sources.

The fuzzy portions of the IAC were implemented using the fuzzylogic library for Python 3 [33]. As an example, the complex fuzzy set for the qualification is displayed in Figure 6. It consists of the calculation of the under- and over-qualification, as well as the final score with its defined weights.

As BPMSs for our prototype, we integrated AristaFlow [34] (using a synchronous service activity approach) and Camunda [35] (using an asynchronous script activity approach). In the following, we focus on the Camunda implementation. It is a well-known application in the BPM context and further provides all required functionality as well as a BPMN Modeler as an open-source solution. In addition to a full implementation of the BPMN 2.0 standard, Camunda also provides a Connector element, allowing easy REST requests from within process instances via script and service activities. The well documented REST-API [36] supports a quick and easy integration of the communication interface.

As the free version of Camunda only provides a BPMS with minimal user management, an extension in form of a minimal REST-Backend (further called CamundaClient) handling users and assignments was required. Users are added via a new backend and saved according to our agent model. The process templates were extended as planned in the solution approach. The utility activity requests a score calculation from the CamundaClient for the subsequent execution activity. This execution activity must contain the Assignment Criteria as described in the activity model. The CamundaClient then loads the required user data from the database and sends a request to the ACE. It is also possible to move this step to the DAC in the ACE; in this case, it would only be required to send the activity ID to the ACE. While we implemented both the synchronous and asynchronous variants, we focus here on the asynchronous one, as it provides additional benefits such as better multi-processing support and should be chosen if

```
def eval_qualification(required: dict, values: dict):
    """eval qualification and over qualification for tasks
    required value and agents value"""
    qualifications = {"types": {}}

    for qualification_type, required_qualification in required.items():
        Qualification = Domain("qualification", 0, 10, res=1)
        qualified = (
            rectangular(required_qualification, 10)
            if required_qualification < 10
            else singleton(required_qualification)
        )
        over_qualified = (
            Set(R(required_qualification, 10))
            if required_qualification < 10
            else Set(constant(0))
        )
        begin_flat = Set(singleton(required))

        if required_qualification != 0:
            begin_flat = MAX(
                ~Set(rectangular(0, required_qualification)),
                Set(singleton(required_qualification)),
            )

        Qualification.qualified = qualified
        Qualification.over_qualified = over_qualified
        Qualification.not_over_qualified = (
            Set(S(required_qualification, 10))
            if required_qualification < 10
            else Set(begin_flat)
        )

        if required_qualification != 0 and required_qualification != 10:
            Qualification.not_over_qualified = product(
                begin_flat, Set(S(required_qualification, 10))
            )

        if required_qualification == 10:
            Qualification.not_over_qualified = begin_flat

        qualifications["types"][qualification_type] = float(
            Qualification.min(values[qualification_type])
        )

    Qualification = Domain("qualification", 0, 1, res=0.001)
    weight = {"electric": 0.25, "computer": 0.25, "social": 0.25, "bio_chemical": 0.25}

    w_func = weighted_sum(weights=weight, target_d=Qualification)
    qualifications["weighted"] = w_func(qualifications["types"])

    return qualifications
```

Figure 6. Code snippet showing fuzzy implementation for qualification.

supported by the utilized BPMS. As soon as the assignment is calculated, the assignment scores are sent from the ACE to the CamundaClient and the assignment in Camunda is handled via the client. Connected to this, the workload of the assigned agent(s) is increased and their Danger Threshold is decreased by the Danger Level of the activity. The Danger Levels can further be reset to the agents' default value, e.g., on a daily or weekly basis as required by labor safety laws.

Alternatively, the IAC could be integrated in the CamundaClient itself, removing the need for the additional REST-Requests between the client and the ACE. The current approach, however, supports the generic usage of the ACE as

```

package com.paradigma.temperature_rules;

Rule "TemperatureLowerThan"
when
|... $sensorData: SensorData()
then
|... $sensorData.setValid(
|... |... $sensorData.getMaxValue()
|... |... > $sensorData.getCurrentValue()
|... );
end

```

Figure 7. Example precondition rule validated by Drools.

a service for multiple BPMS simultaneously and entails less restrictions.

The Rule Interface was implemented for the Drools rule engine [37]. The engine was chosen for its widespread use in the industry as well as its REST interface. While the implementation of the Rule Interface itself has to be customized via an adapter to the utilized rule engine itself, the internal API called by the Assignment Handler was created in a generic way, allowing an easy exchange of rule engines.

Currently the Rule Interface can only be used to filter activities itself. For this, preconditions have to be created in the language of the rule engine and have to be added to the process template. The precondition is then sent to the rule engine when the assignment is requested for the connected activity and it is evaluated. This is available in two modes; loop and single. In loop, the rule is executed in a loop with a small delay between executions until it is fulfilled. In single mode, the condition is evaluated once, if this fails an error is returned to the BPMS. While currently no pre-filtering of agents is supported, this could be added to the Rule Interface in coming updates. Figure 7 shows a simple example for such a precondition rule, validating that the current temperature of a sensor `$sensorData.getCurrentValue` is below the maximal allowed threshold `$sensorData.getMaxValue`.

Powerful multi-processing capabilities were implemented in the Intelligent Assignment Component and managed by an intelligent orchestrator. While the Assignment Component is already realized with a runtime of $O(n)$, its performance can be further increased with our multi-processing approach. Assignments with large numbers of agents can therefore be run in a multi-processing configuration with multiple modes. The default for large requests is $n - 1$ processes, where n is the maximal number of cores available on the machine. This provides maximum calculation speed while still preserving one process for the ACE itself, preventing slowdowns. If the request is too small for multi-processing (the multi-processing overhead would slow down the computation speed), the orchestrator runs the calculation in a single process. Finally, it is possible to run the calculation in $n - m$ processes, where $m, m < n$ is calculated according to the server's performance in multi-processing mode. We implemented a semi-automatic

test setup, calculating the optimal m for a server for 10, 100, 1000, ... 1000000 agents in a single assignment request. The calculated m can then be used in the server configuration to allow maximum performance according to the utilized hardware.

VI. EVALUATION

The evaluation focuses on two aspects of the solution: the first considers performance and scalability implications of our IAC agent assignment algorithm utilizing our prototype within a realistic software and hardware environment, in order to determine if there are unforeseen practical limitations or bottlenecks that would hinder its usage. For the second part, we evaluate assignment optimizations achieved when a BPMS utilizes the IAC versus a BPMS only (Camunda without the IAC) in order to determine if a significant benefit can be shown.

Matplotlib [38] was used to briefly analyze the data as well as to graphically process the results. The evaluation itself was conducted utilizing the Python libraries pandas [39], SciPy [40] and NumPy [41].

A. Performance Evaluation

The performance evaluation was conducted to analyse the performance and scalability implications of our IAC on a virtual server with 90GB of main memory. As an operating system, Debian 10 was chosen utilizing Python 3.7.2 for the algorithm execution. The test was separated in two groups of 10, 100, 1000, 10,000 and 100,000 agents being assigned to a single activity. In the first group, all agents were capable of performing the activity according to the assignment criteria. In the second group, only certain agents were capable of performing the activity. The assignment of each group of agents to their activity was conducted 100,000 times. The groups from 10 to 1000 agents were assigned using the IAC without multi-processing while 10,000 and 100,000 agents were assigned using 17 processes ($n - 1$ mode).

Figure 8 displays the assignment calculation performance if all supplied agents were capable, while Figure 9 displays the calculation performance if only some agents fulfilled the requirements. The calculation duration results show an approximately linear scaling in the single processing mode (10-1000 agents), while multi-processing decreases with larger numbers of agents (10000-100000 agents). Unexpectedly, calculation duration for assignments with only capable agents is lower than that of agents with mixed requirements. This originates from some optimization problems in the elimination of incapable agents. A possible solution to this can be found in the evaluation summary below.

In general, the assignment of high volumes of agents caused no issues for the algorithms. As the IAC is meant to run behind SARs, rule engines, or other performant basic filtering algorithms, a load of 10,000 possible agents for a single activity is further quite unlikely. The runtime in the sub seconds for agent values below 100,000 would also allow the

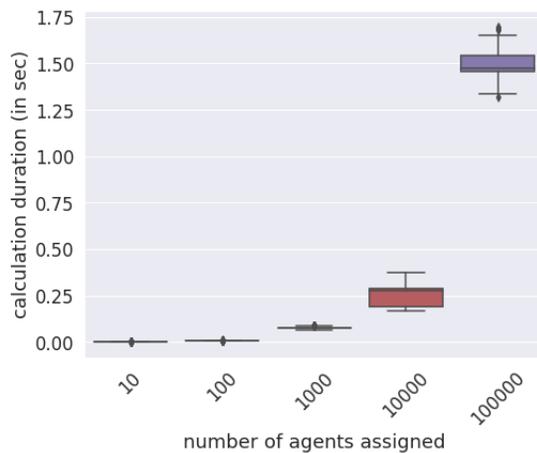


Figure 8. Calculation performance vs. number of capable agents assigned.

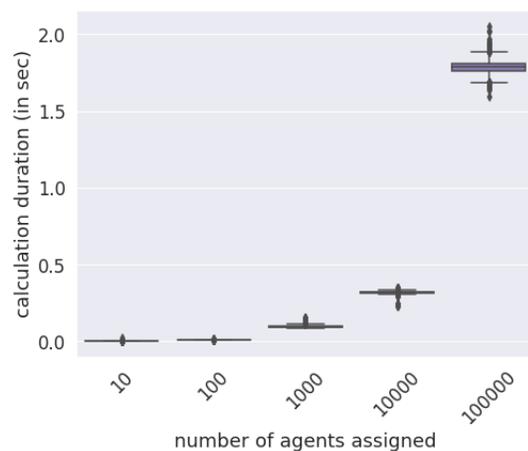


Figure 9. Calculation performance vs. number of capable and non-capable agents assigned.

removal of preliminary filtering, reducing the runtime of the whole BPM process.

B. Integration Evaluation

The integration evaluation was conducted using the AnyLogic simulation software. The AnyLogic simulation was run on a Lenovo T495 with 14GB main memory utilizing Arch Linux as an operating system. No changes were made between the performance evaluation and this one besides the setup of this edition. The laptop and server containing the BPM and ACE were on the same network.

The evaluation was used to compare a BPMS using the IAC against a plain BPMS. To simulate workers and a realistic workflow, an AnyLogic simulation was built and two simulation setups were configured.

A factory with $21,504m^2$ and a total of 29 machines which required maintenance every 16 hours was created. The first maintenance was scheduled between 0 to 16 hours after start of the simulation. Further, the machines had an average breakdown interval of 36 hours. If a machine required maintenance or repair, it started a new Camunda process

instance with the required qualification and the machine's position. The activity takes between 1 to 3 hours and requires an engineering qualification of 4 for maintenance and 6 for repairs. Other qualifications (electric, computer, bio_chemical) were not required and set to 0. A total of 5 agents were available to complete this activity; four internal workers, waiting in a maintenance building in the factory hall and one external agent, waiting 165 meters away. The internal agents had engineering qualifications of 4, 5, 6 and 7 while the external agent had an engineering qualification of 8. The other qualification values were set to 0 to avoid bias. The usage of the external agent was connected to an additional cost of 2500 (25€/activity), while the usage of internal workers incurred no additional costs. In their idle state, an agent checked every 5 minutes if a new activity is available. If they were working, after completion of their current activity they checked if another activity was enqueued. If no activity was enqueued, they switched back to the idle state and moved to their starting position. This part of the setup was identical in both simulation setups.

The factory size and machine breakdown/maintenance intervals were empirically selected to achieve a high utilization of the available agents without overstraining their capacity, allowing for a realistic environment. The qualification of the internal and external assignments were selected in a way, which allows the internal agents to do most of the work on their own, but requiring the external agents support with high workloads. We have refrained from mixing more different tasks/more complex tasks with many different requirements, to provide an easy to understand and analyzable evaluation. Mixing more different task types/requirement constrains would not affect the technical evaluation in a meaningful way, as all fuzzy models are run even with requirements of 0, however, it would become harder to understand and analyse the simulations findings.

In the Camunda Setup (called CMD-Setup), the agents fetched their activities directly from Camunda. All activities of the simulation were available to all of the workers with no further verification. If an activity is available to the group, the agents try to claim it and, if successful, work on it. In the IAC Setup (called IA-Setup) the agents checked their personal worklist at the ACEs REST API. If their personal worklist contains an activity, they start to work on it, otherwise the stayed idle.

A timespan of 36 working hours were simulated for both configurations, using the same seed for the simulations random number generator. This process was repeated 10 times with different seeds to get the statistical relevant test data. For the IAC, the model introduced in Section IV was used. The qualification value was weighted half to increase utilization of the more qualified agents and reduce the downtime of the machines. Further adjustment of the weighting could lead to heavily deviating results. An optimal weighting has to be configured according to the needs of the activities.

Table II shows a general comparison between the CMD (only Camunda, no IAC) and IA simulation (Camunda with the IAC), while III shows a more detailed comparison of internal

TABLE II. IA/CMD-SETUP SIMULATION MEASUREMENTS.

	IA	Camunda
total_activities (amount)	13.98	16.84
work_time (in minutes)	1636.38	1955.20
idle_time (in minutes)	523.62	204.80
cost (in €)	10.00	420.00
avg_over-qualification (value)	0.34	0.09
max_avg_under-qualification (value)	0.00	-0.02
traveled_dist (in meters)	7346.92	8911.79
downtime_maintain (in minutes)	484.18	303.30
downtime_repair (in minutes)	204.00	138.23

TABLE III. INTERNAL/EXTERNAL WORKER SIMULATION MEASUREMENTS.

	IA-int	IA-ext	CMD-int	CMD-ext
total_activities	17.38	0.40	16.85	16.80
work_time	2037.25	32.88	1946.62	1989.50
idle_time	122.75	2127.12	213.38	170.50
cost	0.00	10.00	0.00	420.00
avg_overqual	0.05	1.50	0.06	0.20
max_avg_uqual	0.00	0.00	-0.02	0.00
traveled_dist	9082.78	403.45	8750.17	9558.25

(-int) and external (-ext) worker stats in both simulations. In the following values from Table II will be compared with the more detailed values from Table III.

The average work time and total activities per worker are lower in the IA run, while the utilization of the internal workers (IA-int) is slightly increased and the external utilization (IA-ext) is heavily reduced. The average idle time is increased which can be deducted from the low external utilization. The heavily reduced average cost of a simulation run, if using the IAC instead of a plain BPMS, can be attributed to the preferred use of internal workers.

The increase in over-qualification while using IA instead of plain Camunda can be explained with the low weighting of qualification in the algorithms, as well as the lack of under-qualification in comparison to the CMD-Setup, where under-qualification was generally present. In Table III, the main source of over-qualification in the IA simulation comes from the usage of the external worker, who was mainly used for activities below his qualification. This happened because of an extreme workload and could be solved by employing another internal worker with lower qualification to help out with this activity. This would lead to reduced cost and downtime. Optimization in the simulated company is needed, rather than an adaptation of the algorithm.

The traveled distance for the internal workers is slightly increased in the IA simulation compared to the CMD run. This, however, stands in linear dependency with the increased workload. A stronger weight regarding the distance could reduce this effect.

The downtime in the IA run is around 50% higher than in the CMD-Setup, while the cost was reduced to 4.2% of the CMD-Setup. This was expected behavior, as the algorithms by default try to save money and therefore did not employ the external worker as much as the CMD-setup.

C. Summary

The IAC performed as expected with fast execution times on mid-to-low budget hardware. Scaling was only required for the case when more than 1000 agents could be assigned to the same activity. This is highly unlikely even in companies with more than 1000 employees, as the BPMS most of the time already pre-filters valid agent groups. Further, the number of agents available for assignment in such large corporations could further be reduced by extending the precondition filtering as mentioned in Section V. If further scaling is necessary, it can be readily achieved and works efficiently for at least 100,000 agents per activity. The algorithms further produce comprehensible results for analysis by non-experts, which can be adjusted as required through dynamic weighting of the different variables in the algorithm.

Due to the non-blocking REST-API design and decoupled async assignment process, we do not foresee any multi-tenant performance issues.

Taking the current runtime in the sub seconds for less than 100,000 agents into account, the implementation of agent pre-filtering via rule engines should not be implemented for low- to mid-size systems, as the additional REST calls would ultimately mean a slowdown and increase the overall runtime.

However, the evaluation also shows an unexpected finding, whereby the calculation duration for assignments with only capable agents is lower than that of agents with mixed requirements. As in the second case, the execution criteria are triggered and the final score calculation can be skipped for some agents. This leaves room for optimization regarding the handling of exclusion criteria for future work.

VII. CONCLUSION AND FUTURE WORK

Industry 4.0 stands for highly automated production processes. However, these processes also rely on complicated tasks that can only be performed manually by humans. The integration of such activities into the processes is still problematic. One important issue is efficient task assignment, which is not solved well in contemporary BPMS.

To counteract this, the current contribution described an approach for more effective and efficient activity assignment for Industry 4.0 production processes. The focus of this approach was to build a compact model of fuzzy sets that can be easily applied to real projects, while also combining rules for pre-filtering for more obvious logical determinations and combinatorial constraints that do not require fuzziness. These rules can be easily adjusted and adapted by users and efficiently executed on a rules engine, thus focusing the fuzzy sets on those areas for which it is specialized. For our realization and evaluations we chose a set of important properties that incorporate aspects relevant in current Industry 4.0 production: achieve cost savings by incorporating not only under-qualification but also over-qualification, and the separation between internal and expensive external workers; achieve a balanced workload for all workers to avoid both idle time and overburdened workers; protect the workers from different hazards as enforced by government regulations; and finally,

optimize assignments with knowledge about the locations of workers and their potential activities by minimizing transit overhead.

Besides providing a practical model, our approach also features concepts for the direct integration with BPMS. To demonstrate its feasibility, we have currently implemented, integrated and tested our prototype approach with two concrete BPMS; AristaFlow and Camunda. The approach is built modularly and can be easily expanded. Furthermore, the fuzzy sets used to calculate an assignment utilize weights that can be changed dynamically according to the users' specific needs. It is also possible to use this prototype with any other BPMS supporting BPMN 2.0 with minimal effort.

The evaluation showed that our approach is an efficient way to automatically compute assignments. We evaluated the algorithms regarding performance and built a comprehensive simulation scenario to show its effectiveness and efficiency in providing optimal assignment recommendations. However, the Rule Engine interface, as well as the exclusion criteria filtering of incapable agents, still leaves room for optimization.

For future work, we plan to incorporate a more generic model where not only the weights are dynamic, but also the criterion. Instead of hardcoded values, it should be possible to define them via configuration files, or dynamically as part of the REST call to the ACE. Thus, the approach can be easily adapted for further domains and scenarios by extending or replacing the evaluation criteria. Additionally, we plan to rework the exclusion criteria in order to speed up the removal of unsuited agents from the assignment process. Other upcoming improvements could include utilizing transit path finding algorithms for the distance calculation in order to provide a more realistic and resilient calculation. The duration of activities could also be considered in order to measure the workload not only in the amount, but also in terms of estimated time required to complete the activities.

ACKNOWLEDGMENTS

We thank Felix Gräber for his contribution regarding the fuzzy algorithms and implementation, and Camil Pogolski for his input on the models. This work was partially funded via the PARADIGMA project by "Zentrales Innovationsprogramm Mittelstand" (i.e., the "Central Innovation Programme for small and medium-sized enterprises (SMEs)"), of the Federal Ministry for Economic Affairs and Energy of the Federal Republic of Germany.

REFERENCES

- [1] G. Grambow, D. Hieber, and R. Oberhauser, "Intelligent task assignment in industry 4.0 production processes utilizing fuzzy sets," in *INTELLI 2021, The Tenth International Conference on Intelligent Systems and Applications*, 07 2021, pp. 1–9.
- [2] H. Lasi, P. Fetteke, H.-G. Kemper, T. Feld, and M. Hoffmann, "Industry 4.0," *Business & Information Systems Engineering*, vol. 6, no. 4, pp. 239–242, 08 2014.
- [3] R. Baheti and H. Gill, "Cyber-physical systems," *The impact of control technology*, vol. 12, no. 1, pp. 161–166, 2011.
- [4] D. Karagiannis, "Bpms: business process management systems," *ACM SIGOIS Bulletin*, vol. 16, no. 1, pp. 10–13, 1995.
- [5] C. Moore, "Common mistakes in workflow implementations," *Giga Information Group, Cambridge, MA*, vol. 2, 2002.
- [6] M. Arias, R. Saavedra, M. R. Marques, J. Munoz-Gama, and M. Sepúlveda, "Human resource allocation in business process management and process mining," *Management Decision*, vol. 56, no. 2, pp. 376–405, Jan. 2018.
- [7] M. M. Flood, "The traveling-salesman problem," *Operations Research*, vol. 4, no. 1, pp. 61–75, Feb. 1956.
- [8] L. A. Zadeh, "Fuzzy logic," *Computer*, vol. 21, no. 4, pp. 83–93, 1988.
- [9] G. Grambow, R. Oberhauser, and M. Reichert, "Semantic workflow adaption in support of workflow diversity," in *4th Int'l Conf. on Advances in Semantic Processing*. Xpert Publishing Services, 2010, pp. 158–165.
- [10] G. Grambow, R. Oberhauser, and M. Reichert, "Employing semantically driven adaptation for amalgamating software quality assurance with process management," in *Second Int'l Conf. on Adaptive and Self-adaptive Systems and Applications (ADAPTIVE'10)*. Xpert Publishing Services, 2010, pp. 58–67.
- [11] G. Grambow, R. Oberhauser, and M. Reichert, "Enabling automatic process-aware collaboration support in software engineering projects," in *Communications in Computer and Information Science (CCIS) 303*. Springer, 2012, pp. 73–89.
- [12] S. Kubler, J. Robert, W. Derigent, A. Voisin, and Y. Le Traon, "A state-of-the-art survey & testbed of fuzzy ahp (fahp) applications," *Expert Systems with Applications*, vol. 65, pp. 398–422, 2016.
- [13] V. Shahhosseini and M. Sebt, "Competency-based selection and assignment of human resources to construction projects," *Scientia Iranica*, vol. 18, no. 2, pp. 163 – 180, 2011.
- [14] G. Klosowski, A. Gola, and A. Świć, "Application of fuzzy logic in assigning workers to production tasks," *Distributed Computing and Artificial Intelligence, 13th International Conference*, pp. 505–513, 01 2016.
- [15] H. Seifi, N. Shams, and K. M. Cyrus, "A novel self-regulating and intelligence meta-heuristic-fuzzy approach for integrated and optimal human resource allocation in normal and critical conditions," *International Journal of Fuzzy Systems*, vol. 24, no. 1, pp. 121–134, Jun. 2021.
- [16] K. Kluza and G. J. Nalepa, "A method for generation and design of business processes with business rules," *Information and Software Technology*, vol. 91, pp. 123–141, Nov. 2017.
- [17] D. Antonelli and G. Bruno, "Dynamic distribution of assembly tasks in a collaborative workcell of humans and robots," *FME Transactions*, vol. 47, pp. 723–730, 01 2019.
- [18] X. Xu, J. Hao, L. Yu, and Y. Deng, "Fuzzy optimal allocation model for task–resource assignment problem in a collaborative logistics network," *IEEE Transactions on Fuzzy Systems*, vol. 27, no. 5, pp. 1112–1125, 2019.
- [19] E. Simpson and S. Roberts, *Bayesian Methods for Intelligent Task Assignment in Crowdsourcing Systems*. Cham: Springer International Publishing, 2015, pp. 1–32.
- [20] J. Quinzaños, A. Cartas, A. P. Vidales, and A. Maldonado, "iDispatcher: Using business rules to allocate and balance workloads," *Frontiers in Artificial Intelligence and Applications*, vol. 261, pp. 110–119, 2014.
- [21] D. Li, G. Wang, C. Huang, S. Liu, and W. Zhang, "The research on auto-assignment method of service orders based on rule reasoning," in *2019 IEEE 4th Advanced Information Technology, Electronic and Automation Control Conference (IAEAC)*, vol. 1, 2019, pp. 2694–2698.
- [22] G. Havur, C. Cabanillas, J. Mendling, and A. Polleres, "Resource allocation with dependencies in business process management systems," in *Lecture Notes in Business Information Processing*. Springer International Publishing, 2016, pp. 3–19.
- [23] R. Sikal, H. Sbai, and L. Kjiri, "Promoting resource discovery in business process variability," in *Proceedings of the 2nd International Conference on Networking, Information Systems & Security*, ser. NISS19. New York, NY, USA: Association for Computing Machinery, 2019.
- [24] J. Erasmus, I. Vanderfeesten, K. Traganos, X. Jie-A-Looi, A. Kleingeld, and P. Grefen, "A method to enable ability-based human resource allocation in business process management systems," in *The Practice of Enterprise Modeling*, R. A. Buchmann, D. Karagiannis, and M. Kirikova, Eds. Cham: Springer International Publishing, 2018, pp. 37–52.
- [25] O. Vasilecas, D. Kalibatiene, and D. Lavbič, "Rule- and context-based dynamic business process modelling and simulation," *Journal of Systems and Software*, vol. 122, pp. 1–15, Dec. 2016.
- [26] S. Ihde, L. Pufahl, M.-B. Lin, A. Goel, and M. Weske, "Optimized resource allocations in business process models," in *Lecture Notes in*

- Business Information Processing*. Springer International Publishing, 2019, pp. 55–71.
- [27] W. Tan, L. Zhao, N. Xie, A. Tang, X. Hu, and S. Tang, "Methods for optimal resource allocation on cooperative task scheduling in cross-organizational business process," in *Computational Data and Social Networks*. Springer International Publishing, 2018, pp. 126–138.
- [28] *Business Process Model and Notation(BPMN) - Version 2.0.2*. Object Management Group, 12 2013, retrieved: 2021.06.10. [Online]. Available: <https://www.omg.org/spec/BPMN/2.0.2/PDF>
- [29] T. Allweyer, *BPMN 2.0: introduction to the standard for business process modeling*. BoD–Books on Demand, 2016.
- [30] G. Grambow, D. Hieber, R. Oberhauser, and C. Pogolski, "A context and augmented reality bpmn and bpms extension for industrial internet of things processes," in *9th International Conference on Business Process Management (BPM 2021) Workshop Proceedings*. Springer, 2021.
- [31] G. Grambow, D. Hieber, R. Oberhauser, and C. Pogolski, "Supporting augmented reality industry 4.0 processes with context-aware processing and situational knowledge," in *eKNOW 2021*, 07 2021, pp. 29–36.
- [32] G. C. Hillar, *MQTT Essentials-A lightweight IoT protocol*. Packt Publishing Ltd, 2017.
- [33] A. Kiefner. Python3 fuzzylogic. Last visited: 2021.06.14. [Online]. Available: <https://github.com/amogorkon/fuzzylogic>
- [34] M. Reichert, "Enabling flexible and robust business process automation for the agile enterprise," in *The Essence of Software Engineering*. Springer, Cham, 2018, pp. 203–220.
- [35] Camunda. Last visited: 2021.06.10. [Online]. Available: <https://camunda.com>
- [36] Camunda docs - rest api reference. Last visited: 2021.12.15. [Online]. Available: <https://docs.camunda.org/manual/7.5/reference/rest/>
- [37] M. Proctor, "Drools: A rule engine for complex event processing," in *Applications of Graph Transformations with Industrial Relevance*. Berlin, Heidelberg: Springer Berlin Heidelberg, 2012, pp. 2–2.
- [38] J. D. Hunter, "Matplotlib: A 2d graphics environment," *Computing in Science & Engineering*, vol. 9, no. 3, pp. 90–95, 2007.
- [39] T. pandas development team, "pandas-dev/pandas: Pandas," Feb. 2020.
- [40] P. Virtanen *et al.*, "SciPy 1.0: Fundamental Algorithms for Scientific Computing in Python," *Nature Methods*, vol. 17, pp. 261–272, 2020.
- [41] C. R. Harris *et al.*, "Array programming with NumPy," *Nature*, vol. 585, no. 7825, pp. 357–362, Sep. 2020.

ARPF - an Augmented Reality Process Framework for Context-Aware Process Execution in Industry 4.0 Processes

Gregor Grambow, Daniel Hieber, Roy Oberhauser and Camil Pogolski

Dept. of Computer Science

Aalen University

Aalen, Germany

e-mail: {gregor.grambow, daniel.hieber, roy.oberhauser, camil.pogolski}@hs-aalen.de

Abstract—Although production processes in Industry 4.0 settings are highly automated, many complicated tasks, such as machine maintenance, continue to be executed by human workers. While smart factories can provide these workers with some digitalization support via Augmented Reality (AR) devices, these AR tasks depend on many contextual factors, such as live data feeds from machines in view, or current work safety conditions. Although currently feasible, these localized contextual factors are mostly not well-integrated into the global production process, which can result in various problems such as suboptimal task assignment, over-exposure of workers to hazards such as noise or heat, or delays in the production process. Current Business Process Management (BPM) Systems (BPMS) were not particularly designed to consider and integrate context-aware factors during planning and execution. This paper describes the AR-Process Framework (ARPF) for extending a BPMS to support context-integrated modeling and execution of processes with AR tasks in industrial use cases. Our realization shows how the ARPF can be easily integrated with prevalent BPMS. Our evaluation findings from a simulation scenario indicate that ARPF can improve Industry 4.0 processes with regard to AR task execution quality and cost savings.

Keywords—Business Process Management Systems; Augmented Reality; Fuzzy Logic; Business Process Modeling Notation; Resource Assignment Automation.

I. INTRODUCTION

Today's manufacturing industry heavily relies on smart factories which enable a better customer orientation as well as more efficient and individual production. However, despite the focus on a high automation level and the utilization of autonomous systems, human involvement in complex processes still plays a crucial part. Human workers often have to make important decisions or perform complex tasks, such as machine maintenance.

This paper extends our previous work [1], where we introduced ARPF, that aims to holistically integrate AR worker-based and production processes, utilizing Industry 4.0 smart factories with their Cyber-Physical Systems (CPS) [2], leveraging their wide range of sensor capabilities to provide context-based AR support for human tasks. This not only provides support to the worker itself, during task execution, but also enables comprehensive optimization of production with regard to criteria, such as costs, resource consumption, quality or availability.

While using AR devices to support complex tasks executed by humans is no novelty, the integration of such activities in

the global production process remains a challenge. A primary reason for this is that human AR tasks depend on a large number of factors that are not typically represented in the overall higher-level business and production process. This includes the following factors:

- The AR tasks rely on different contextual data sets, e.g., external information sources supporting task execution, such as maintenance manuals, alternative procedures, checklist variability, live data from external systems or sensors of machines, the task executor and their decisions, and context-sensitive AR data such as the relative position of the worker or the machine.
- For maximal effectivity and efficiency, the task must be assigned to the most suitable worker. Simple Staff Assignment Rules (SARs) of contemporary BPMS governing the production processes are only capable of determining if an agent is able to perform a task, but not their level of suitability. For AR tasks in complex Industry 4.0 settings, however, many parameters should be taken into account, such as the position of the worker and the task, the qualification of the worker, or the workload of each worker. Otherwise, task execution might be suboptimal or too expensive, e.g., because of overqualification of the worker or long distances between him and the task. Furthermore, worker safety is usually enforced by government regulations and workers' exposure to hazards such as heat, noise, and danger must be taken into account.
- Usually, workers processing AR tasks are able to communicate via the AR device. However, as the AR tasks are not integrated with the global process, decisions or information provided by the worker cannot be directly utilized in that process, leading to delays or incorrect activity choices.

Contemporary BPMS lack facilities for representing and exploiting such data sets as well as contextual factors. Usually, these systems utilize standard BPM languages such as the Business Process Modeling Notation (BPMN) [3], which were not designed to integrate such information into the process templates. Subsequently, live data and situational knowledge cannot be readily utilized in the process instances based on such process templates.

In prior work we developed an approach for contextual process management [4][5][6] which was tailored towards software engineering processes and did not involve the complex specifics of Industry 4.0 nor AR processes. To overcome the aforementioned limitations, we contribute ARPF, an integrated framework extending current BPMS with the following features:

- 1) Facilities to model processes that incorporate contextual factors applicable to human AR tasks.
- 2) Incorporating real-time context data in BPM processes to enable automated context-dependent decision and execution support.
- 3) An interactive AR activity interface for such processes, enabling bi-directional communication between the process and the AR-supported worker.
- 4) An intelligent task assignment component capable of utilizing contextual data for fine-grained suitability levels, able to optimally assign workers to tasks.
- 5) Easy extension of existing BPMS.

This paper extends our previous work [1] with an expanded description providing further AR details and extending the evaluation and related work.

The remainder of this paper is structured as follows: Section II describes the concept and solution approach, while Section III provides realization details. Thereafter, Section IV evaluates the technical capabilities of the implemented system and the empirical results of its usage with AR users. Section V elaborates the background of the research as well as related work. Finally, Section VI provides a conclusion and outlook on future work.

II. SOLUTION APPROACH

This section describes our concept for a context-aware system with AR support, called AR-Process (ARP) Framework (ARPF). It is conceived as a generic extension that any BPMS can readily integrate, providing facilities for representing contextual and AR information in executable processes in conjunction with an enactment component.

A. Contextual Processes

To enable the application of the ARPF in both new and existing processes and enable easy integration in any BPMS, the contextual information will be integrated into the processes via a generic BPMN 2.0 extension. Extending the BPMN standard not only allows an easy integration (requirement 2), but also allows the reuse of the existing BPMN service and script activities [7], heavily reducing implementation efforts. Such activities provide an intuitive interface between the BPMS and the ARPF. With this approach it is possible to decouple the ARPF from the process itself and provide it as a service to any BPMS supporting BPMN 2.0.

In the following, we will elaborate on the context data and rules or conditions crucial for contextual ARP execution. The context is separated into three major parts: global, process, and activity. A model can be seen in Figure 1.

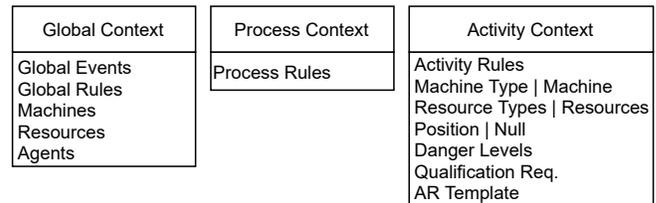


Figure 1. Context data model.

The *global context* represents a cross-process entity containing all required global information. This includes information about different entities external to the BPMS. In particular, all machines, resources, and available agents. Further, the global context should provide facilities for defining conditions regarding the context that must be verified and fulfilled before an activity can be executed (requirement 2), e.g., check if an agent has the required danger clearance for an activity. This is realized by a *global rule set*. Another important factor is external context information that must be provided to the ARPF, e.g., priority changes for customer orders. Such data is incorporated via a global event system. In this manner, real-time context integration into BPM processes on a global level (requirement 2) can be achieved.

In addition to such global information, each process type may also have specific contextual conditions, e.g., if a specific process should only access a subset of the available machinery or use only special types of tools. To achieve this, a *process context* is employed that can overlay applicable portions of the global context if required. It further contains an additional *process rule set*. The latter is similar to the global rule set, but is limited to the processes of this type.

The third important entities requiring contextual information are concrete activities. To support these, an *activity context* is defined. It contains specific information for a single activity of a specific activity type and can be further specified during the ARP execution. As for the process and global context, an *activity rule set* is present to enable fine grained conditions on the activity level. On this level, however, a set of additional contextual information is required to enable an efficient assignment of the best suitable worker for each task. This incorporates data such as the danger levels the task may involve, the qualification to successfully complete it (both defined as a dynamic set of key value pairs, containing values between 0 to 1), and the position of the activity represented by a three-dimensional vector X, Y, Z. Machine types can also be defined, as well as additional resources required for the activity, allowing the inclusion of machine context data directly into the process. Finally, the information, which AR-Component should be displayed to the worker while executing the activity (requirement 3) must be present. This is achieved by the AR Template.

B. Data Models

In addition to the contextual information added to the processes that governs how activities should be executed effi-

Resource Model	Machine Model	User Model
Position Danger Levels Qualification Req.	Position Danger Levels Qualification Req. Sensors	Danger Thresholds Position Qualification Assignment Cost Utilisation

Figure 2. Actor models.

ciently, the ARPF also requires information about the physical entities involved in process execution. In particular, three entities are crucial: the workers, the machines where activities (e.g., a maintenance task) are executed and their position, and resources required for such activities (e.g., materials, tools). To provide such information to the ARPF, three models are created, which can be found in Figure 2.

As simple BPM engines do not provide entries for resources and machines, new models must be created. Both contain a position, connected danger levels (e.g., noise with machines or chemical hazards with resources), and the required qualification to safely and efficiently work with the machine/resource. Machines usually also contain sensors providing real-time information about important production parameters. These are also included in the model.

Finally, the BPMS user/agent concept has to be extended, as current BPMS lack sufficient information to support AR activities intelligently. In order to assign agents to activities, the BPMS must possess compatible models. This can be achieved by extending the agent model of a BPM engine with values for position, qualification, and danger thresholds. Further, in most cases, cost-effective activity execution is also a requirement. Therefore, we incorporate information about the additional cost of an assignment of this agent (e.g., salary of an external worker, or weekend surcharge if it is not part of the contract) and the current utilization of the worker to avoid unbalanced workloads.

C. Process Modeling

In current BPMSs, there are rather limited and generic facilities to add context data to processes. This concerns the process modeling tools as well as the processes themselves. To overcome these limitations, our approach for adding contextual data for ARP execution is to realize such datasets as an extension for the most prevalent process language currently, BPMN 2.0. That way, the integration in a BPMS can be readily achieved, as any BPMN 2.0 compatible modeling tool can be easily extended (cf. requirement 1). To show the feasibility of this approach, we implemented a prototype extension integrated into a prevalent BPM modeling tool. Further details can be found in [8].

With an extended modeling tool at hand, a process engineer can add all contextual information and dependencies crucial for ARP execution support to new as well as existing process models without programming knowledge. Users, machines, and additional resources can be specified, including relevant parameters such as their position. With appropriate data struc-

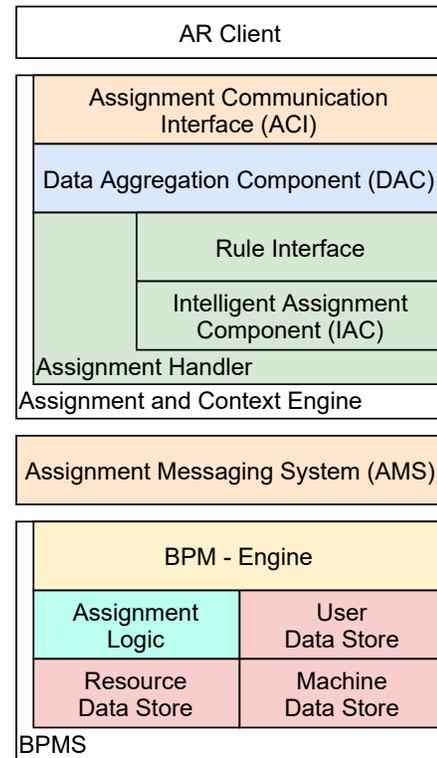


Figure 3. ARPF concept architecture.

tures in place, relevant live data (e.g., from machine sensors) can be incorporated in the processes as they are executed. This data, in turn, is utilized by the components of the ARPF to provide more efficient task assignments and more effective support of the AR activities in the process.

D. AR - Process Enactment

The core architecture of the ARPF for contextual ARP enactment is depicted in Figure 3.

The core component of ARP process enactment is the Assignment and Context Engine (ACE). To provide a generalized and independent solution, this component is decoupled from the utilized BPMS. This permits a finer engineering of the ACE independent of the utilized suite. To achieve this, the ARPF incorporates two language- and platform-neutral generic communication components. The Assignment Communication Interface (ACI) enables communication between the ACE and both the BPMS, as well as the client software on the AR devices, while the Assignment Messaging System (AMS) manages live data from the ARPF environment. That way, the ACE can be realized independent of any preexisting programming language or BPMS limitations. This allows the usage of the ARPF with a wide range of existing BPMS (requirement 5).

Many BPMS are provided as a standalone BPM engine (e.g., Camunda [9], jBPM) and therefore require external software to build a fully functional BPMS capable of managing all crucial data sets for contextual process enactment. To overcome these limitations and provide an easy way to extend BPM engines,

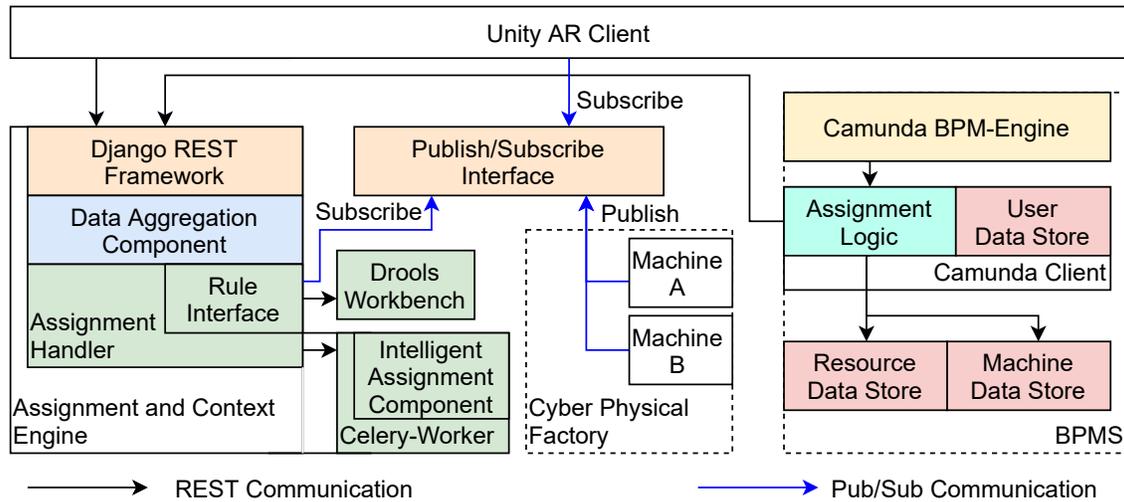


Figure 4. ARPF camunda implementation.

we provide three generic Data Stores (DS): a User-DS, a Resource-DS, and a Machine-DS. These contain additional context information as specified by the aforementioned data models (cf. Figure 2), such as a more refined user model, information about all machines used in the factory, and resources required to complete tasks. This extended context data is required in the assignment process and during the activity execution in the AR-Client.

The Assignment Logic Component (ALC) of the BPM engine is used as a bridge between the engine and ACE. It aggregates all required context data for assignments; however, it is also possible to integrate the assignment request completely in the process itself via service or script tasks defined in the BPMN 2.0 standard [7] (requirement 5).

If an assignment request is sent to the ACE via the communication interface, the request is forwarded to the Data Aggregation Component (DAC) and validated for completeness. If some required context data is missing, the DAC will request it from the corresponding DS. Afterwards, the assignment request is forwarded to the Assignment Handler. The handler can then calculate a specific assignment score for the requested activity and agents in the Intelligent Assignment Component (requirement 4). If required, a presorting can be applied in a rule engine via the Rule Interface. Further preconditions of assignments (e.g., only assign the task if a sensor value is below a certain threshold) can be handled by the rule interface. To guarantee an optimal fine granular assignment score calculation fuzzy sets are utilized [10]. In this case, these are to be preferred to other solutions like Machine Learning (ML) and chaining. In contrast to ML approaches, with fuzzy sets no preexisting data sets are required nor is a training phase required, as weights can be defined directly and transparently according to the user's own knowledge and experience. Further, a fine granular calculated score between 0 and 1 is possible instead of the simple true or false of a chaining approach. Further details on our algorithm and its implementation can be found in [11].

The final component of the ARPF is the AR-Client. The latter should be implemented as generic as possible to be available to a wide range of AR devices, e.g., tablets, goggles and even smartphones. The client is able to request all relevant process data via the ACI, and activities can be started, executed and completed in the AR-Client without the need to change to another software client e.g., a PC-interface, or web-client (requirement 3). Thanks to the provided AMS, it is further possible to consume real-time context changes on multiple levels (e.g., a global change of activity priorities or sensor data send from a machine connected to the activity being executed) (requirement 2).

III. REALIZATION

This section describes the technical realization of the ARPF. It further details the communication between the components. While this section describes its integration with Camunda as a BPM engine and the AristaFlow BPM Suite [12] to demonstrate its capabilities with two mature and prevalent BPMS, the framework can be used with all BPM-Engines supporting REST-calls or external code execution either via extensions or script tasks. The provided AR-Client can further be used with a majority of current AR-devices.

The prototype was implemented using Python due to its rapid prototyping capabilities and large spectrum of available libraries. As a base image for the ACE, a Django server was used which can be readily scaled for production deployment. To implement the ACI, the Django REST framework was integrated, providing a REST interface on top of the Django service. For the AMS, handling the real-time machine sensor communication, the Publish/Subscribe (Pub/Sub) system MQTT [13] was chosen, utilizing the Eclipse Mosquitto broker as the main component. As both technologies use well-defined industrial standards, an easy integration in BPMS is supported.

Figure 4 shows the architecture for the implementation of the ARPF with Camunda. Compared to the concept from Figure 3, some minor changes were made and the communication

specified. The implementation of the AristaFlow BPM-Suite follows the same base architecture; however, the full suite is provided by AristaFlow, removing the need for our own Data-Stores or an ALC. The communication is symbolized by colored arrows in Figure 4.

While the AristaFlow Suite does not require extensions, the Camunda solution requires implementation of a minimal BPM-Suite around the engine itself. This could either be realized as a single Java application relying on the Camunda Java API or using REST. In order to stay consistent with the general architecture, we use REST for our minimal BPM-Suite and split it into three sections. The Camunda BPM engine in its base version, a Camunda Client Django server containing the Assignment Logic, and the User-DS as well as a final Django Server containing the Resource and Machine Data Store. In order to connect the ARPF to a BPMN process template, it is required to create a service or script task sending a REST call to the Camunda Client. This call must contain the process instance id that can be acquired during process runtime in the same activity. During the process execution, the Camunda engine then calls the Assignment Logic via the created activity and triggers the assignment process. The Assignment Logic confirms the request to the engine and then spawns a new process handling the request. It then aggregates all data required for this assignment and sends an assignment request to ACE.

The Django REST Framework based ACI receives the assignment request and then executes the data aggregation component, validating that all required data for an assignment is available. In the Camunda implementation all required data is already present, in the AristaFlow implementation, the required data can be received from predefined endpoints. Afterwards the Assignment Handler is called. If preconditions are implemented (e.g., confirming the temperature of a machine sensor), the Rule Interface takes action. It first subscribes to all required machine sensor data endpoints via the Mosquitto Broker and then calls the connected rule engine via REST. In the implementation Drools [14] is used for the Camunda Implementation while AristaFlow provides its own XPath based solution. The preconditions can either be run in a loop (e.g., waiting for a sensor to cool down) until the condition is fulfilled, or in single-shot mode, aborting the assignment if the check is negative.

If the assignment is aborted, a response is sent to the Camunda Client/AristaFlow suite which are then required to provide a fallback plan, e.g., a retry after some time, a fallback process, or human intervention.

If the preconditions have been fulfilled, the assignment request is forwarded to the Intelligent Assignment Component (IAC), which itself is detached from the ACE to a celery worker. Utilizing the Celery Python framework, all assignment calculations are outsourced from the ACE and do not bind resources, therefore the I/O operations of the backend are not affected, even if many concurrent assignments are calculated. Each fuzzy assignment calculation is assigned its own processor for optimal execution speed. After the

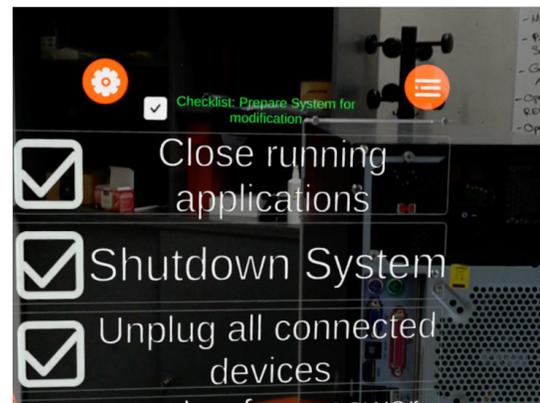
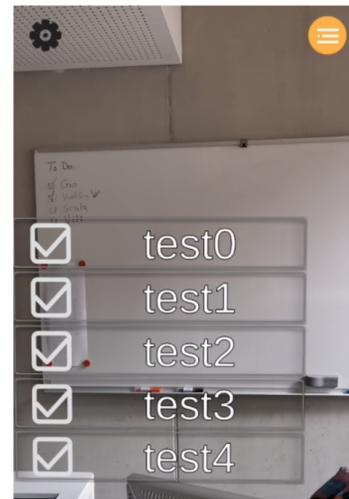


Figure 5. AR-Client, top table/mobile phone version, bottom Magic Leap 1 version.

calculation is finished the IAC sends the assignment to the Celery Client/AristaFlow Suite handling the assignment update in the BPM engine.

The AR-Client is implemented using the Unity AR Foundation framework; this allows the creation of a generic AR frontend usable with a majority of present AR devices such as AR goggles, tables, or phones. Figure 5 shows an early version for the tablet/mobile phone user interface, as well as a later version for the Magic Leap 1. Instead of communication with the BPM-Suite itself, the AR-Client communicates via REST with the ACE and all requests to other sources are handled by the ACE. This enables the creation of a truly generic frontend independent of the BPMS, as all requests are parsed to the required model in the ACE. With this approach combined with a powerful AR interface, the user is able to complete and perform all activities in the AR-Client without the need to utilize another software solution or device. As the BPM workflow is still handled solely by the BPMS, it is however possible to switch at any moment to another solution (e.g., the Camunda Tasklist or the AristaFlow Client) if the worker deems it more beneficial, e.g., filling a long form or accessing specific resources.

While all process management communication is handled

via REST between the AR-Client and ACE, the client can also access the Pub/Sub data via the Pub/Sub interface. It is therefore possible to see all relevant sensor values of a machine while working on it, or receive global updates (e.g., a change of priority or information a new assignment).

The final component of the ARPF is the Pub/Sub Interface, handling all MQTT messages. This contains all machines sensor data for the Rule Interface or the AR-Client, as well as global worker specific updates such as a new assignment or priority updates. While the Camunda Client makes no use of global events via MQTT, the event feature is implemented in the AristaFlow suite.

In our prototype a Cyber Physical Factory is simulated using the OPC-UA protocol to connect machines' sensor data to the ARPF. As OPC-UA supports MQTT, this can be achieved in an easy and generic way, further easing the implementation into existing production environments.

To enable the creation of context-aware processes, a new BPMN modeler is created as an extension of the open source Camunda Modeler. While it is possible to create processes using the ARPF with any BPMN 2.0 modeler, a specific implementation comes with certain advantages. The modeler is linked to the different data stores and can therefore display all available machines, resources, and workers as specific entities or groups (e.g., CNC mill, maintenance workers, etc.) during the modeling of processes. This allows the process engineer to easily include the context during process creation. Further, it is possible to see available rules of the connected rule engine, enabling their integration as preconditions to activities. Moreover, the ARPF specific assignment request is moved to the background and no longer visible in the BPM template as a separate task. Removing them from the user visibility greatly reduces the potential for an overloaded user interface and directs the focus on the more relevant elements.

IV. EVALUATION

ARPF was evaluated following a bipartite approach. In the first evaluation, ARPF technical capabilities were evaluated in a simulation environment. This approach was chosen over a real factory test environment for the benefit of a safe and more controlled environment, easy reproducibility, and providing a large set of test runs. The second part of the evaluation consisted of an empirical evaluation focusing on the AR interface.

Both evaluations used a test setup integrating Camunda as the BPM engine and Drools as the rule engine.

A. Simulation of Worker Activities in an Industry 4.0 Setting

The complete framework was deployed on a virtual Linux server with 90GB main memory. However, the memory consumption never exceeded 24GB during our evaluation and can easily be halved by removing the Drools rule engine. The AnyLogic simulation was run on a Lenovo T495 with 14GB main memory utilizing Arch Linux as an operating system. To simulate values for the machine sensor, an OPC-UA server

was hosted, utilizing a common industrial standard for this use case.

The evaluation was used to compare a BPMS using the ARPF against a plain BPM engine. To simulate workers and a realistic workflow, an AnyLogic simulation model was created and two simulation setups were configured.

As an environment, a factory with $21504m^2$ and a total of 29 machines requiring maintenance every 16 hours were created. The first maintenance was scheduled between 0 to 16 hours after start of the simulation. Further, the machines had an average breakdown interval of 36 hours. If a machine required maintenance or repair, a new Camunda process instance with the required worker qualification and the machine's position was started. The activity takes between 1 to 3 hours and requires an engineering qualification of 4 for maintenance and 6 for repairs. Other qualifications (electric, computer, bio_chemical) were not required and set to 0. As most modern manufacturing environments contain hazards requiring special training and regulations dangers were implemented in the simulation represented by values for noise: 0.01, heat: 0.03, electricity: 0.05, and chemicals: 0.02. While these values are quite abstract, they can easily be further refined and specified. A total of 5 workers (the agents in this use case) were available to complete these activities. Four internal workers, waiting in a maintenance building in the factory hall and one external worker, waiting 165 meters away. The external worker is used to display the need for highly trained personal which often has to be contracted by external service providers. The internal workers had engineering qualifications of 4, 5, 6 and 7 while the external worker had an engineering qualification of 8. The other qualification values were set to 0 to avoid bias. Their danger thresholds were set to 0.7 for all values. The usage of the external worker further was connected to an additional cost of 25000 (250€/activity), while the usage of internal workers incurred no additional costs. In their idle state, a worker checked every 5 minutes if a new activity was available. If they were working, they immediately checked after completion of their current activity for another enqueued activity. If no activity was enqueued, they switched back to the idle state and moved to their starting position. The simulation was separated into 5 work-shifts (each 8 hours long) with a break of 4 hours between shifts. During this break, workers were allowed to complete their current activity, but could not start new ones nor was it possible for machines to create a new task during the break. At the beginning of each work-shift, all tasks are reassigned and the danger thresholds of workers are reset to their default.

In the Camunda Setup (called CMD-Setup in the following), the workers fetched their activities directly from Camunda. All activities of the simulation were available to all of the workers and no further verification performed. If an activity is available to the group, the workers try to claim it and, if successful, work on it. In the ARPF Setup (further called ARP-Setup), the workers checked their personal worklist at the Assignment Engine REST API. If their personal worklist contains an activity, they start to work on it, otherwise they

TABLE I. ANYLOGIC ARP EVALUATION.

	ARP	Camunda
work_time	2103.31	2310.60
idle_time	524.49	396.38
avg_overqual	0.12	0.08
avg_tasks_day	3.52	3.62
violations	0.00	5.12
traveled_distance	9304.40	9502.27
cost	2000.00€	4600.00€
max_avg_underqual	0.00	-0.02
downtime_maintain	439.83	293.14
downtime_repair	218.90	249.32

TABLE II. ANYLOGIC WORKER EVALUATION.

	ARP-int	ARP-ext	CMD-int	CMD-ext
work_time	2358.93	1080.83	2336.76	2205.95
idle_time	324.34	1325.07	381.82	454.65
avg_overqual	0.04	0.42	0.05	0.19
cost	0.00	200.00	0.00	460.00
avg_tasks_day	4.00	1.60	3.60	3.68
violations	0.00	0.00	4.88	6.10
traveled_distance	9952.32	6712.70	9386.80	9964.14
max_avg_underqual	0.00	0.00	-0.02	0.00

remained idle.

The five workdays were simulated for both configurations, using the same seed for the simulations random number generator. This process was repeated 10 times with different seeds to get statistically relevant test data. For the ARPF the model introduced in Section III was used. The qualification value was weighted half, to increase utilization of the more qualified workers and reduce the downtime of the machines. Further adjustment of the weighting could lead to heavily deviating results. An optimal weighting has to be configured according to the needs of the activities.

Table I shows a general comparison between the CMD and ARP simulation, while Table II shows a detailed comparison of internal and external worker stats in both simulations. In the following, values from Table I will be discussed and argued with the values from Table II.

The average work time and total activities per worker are lower in the ARP run, while the utilization of the internal workers (ARP-int) is slightly increased and the external utilization (ARP-ext) is heavily reduced. The average idle time is increased, which results from the low external utilization. The heavily reduced average cost of a simulation run, if using the ARPF instead of a plain BPM engine is due to the preferred use of internal workers. The increase in overqualification while using ARP instead of plain Camunda can be explained with the low weighting of qualification in the algorithms and no under-qualification, in opposition to the CMD-Setup, where under-qualification was generally present (to make it more realistic, under-qualified workers required 60 minutes longer than qualified workers). Taking a look at Table II, the main source of overqualification in the ARP simulation comes from the usage of the external worker, who was mainly used for activities below their qualification. This happened because the workload was too high and could be resolved by employing another internal worker with lower qualification to help out

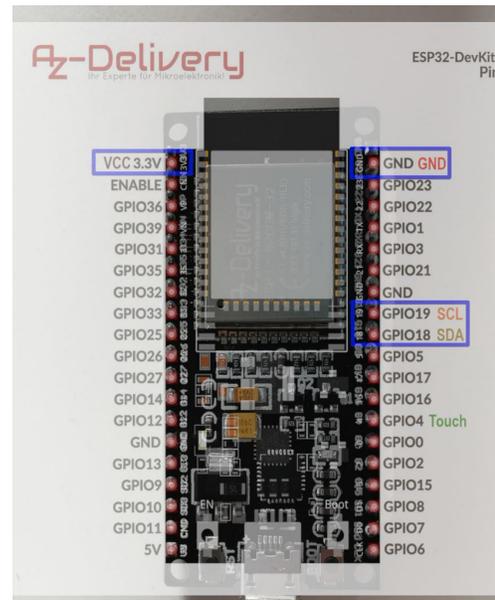


Figure 6. ESP32 with AR overlay.

with these activities. This would lead to reduced costs and downtime. Optimization in the simulation or company values is needed rather than an adaptation of the algorithm.

The traveled distance for the internal workers is slightly increased in the ARP simulation compared to the CMD run. This correlates with the increased workload, and a stronger weight regarding the distance could reduce this effect. While the time for maintenance in the ARP run is around 40% higher than in the CMD-Setup, the actual down time for repairs could be reduced. This would increase the overall efficiency, as machines scheduled for maintenance still function properly while fast intervention is required on broken down machines. Further, the cost could be reduced to 43% of the CMD-Setup.

While the ARPF also utilized rules via Drools to validate if the work on the machine was safe by checking the values of the machine's temperature sensor against a max threshold, the base BPM engine did not provide such features. Violations against this precondition can be found under violations in Table II. In a real environment this would either lead to a safety regulation violation or would require a change of tasks for the worker, leading to even lower performance.

Finally, the ARPF could support workers more efficiently with their tasks, as it displays AR-instructions according to the qualification of the user. This could lead to a further speedup which has to be evaluated in a real-world setup.

Concluding, the ARPF worked as expected and the IAC produced comprehensible results. The utilization of ARPF in the simulations reduced the downtime of machines through failure and prevented any safety regulation violations.

B. Empirical Evaluation with AR Users

In order to gather empirical insights about the AR interface while guaranteeing compliant execution regarding the covid restriction present at the time, we conducted the AR evaluation

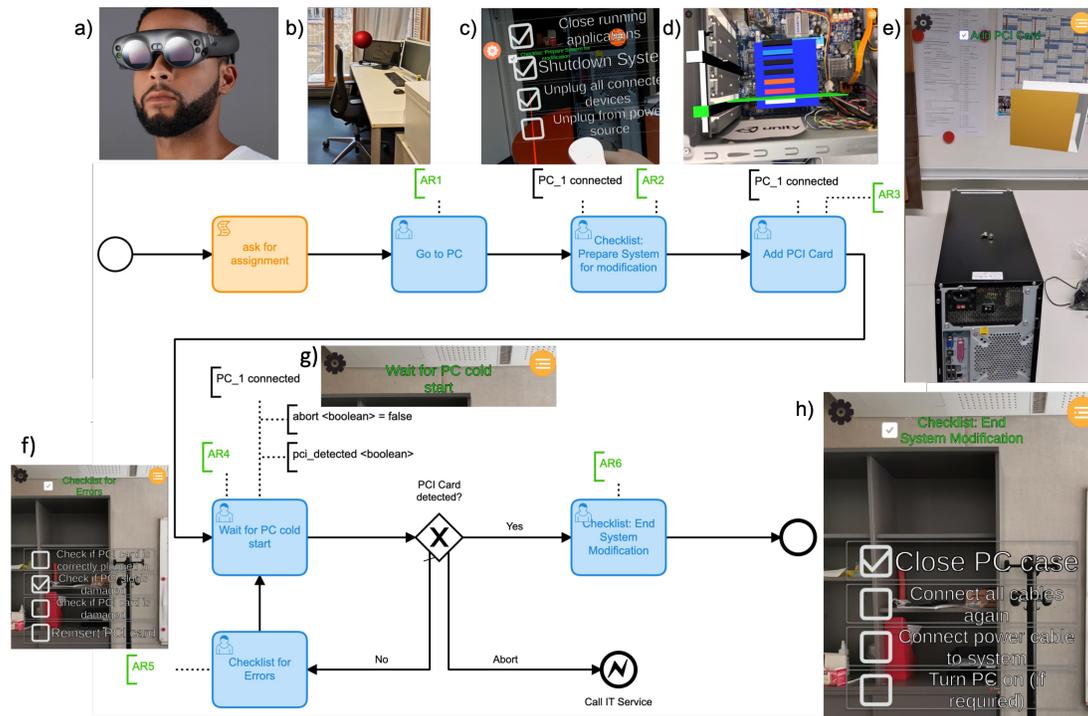


Figure 7. AR evaluation workflow display.

with 22 computer science and electrical engineering bachelor students in the semesters 2-8. Of these subjects, only 3 had advanced experience with AR devices, seven had no prior AR experience at all, ten had used an AR device before (two of which had used AR devices multiple times but possessed no advanced AR usage competency). The evaluation consisted of two use cases which had to be completed with AR support and with instructions on paper, one half starting with paper, the other half with AR. The two tasks were:

- 1) Connecting an ESP32 controller to a display
- 2) Installing a PCI card in a tower PC

In our pre-evaluation, the usage of a handheld AR device (phone or tablet) proved to be a hindrance in this use cases, as the AR device's camera had to be focused on the object to provide guidance, while, at the same time, both hands were required to perform the tasks efficiently. To overcome this hindrance, we selected a smart goggle (Magic Leap 1) for the final evaluation. With this approach the subjects could move the target-object into focus by looking at it, enabling the AR overlay, and had both hands at their disposal at the same time.

In use case 1 AR was primarily used to support the subjects by providing an overlay on top of the ESP32 controller, highlighting the required pins (cf. Figure 6). Further, the subjects were able to work through the BPM process only using the AR device, completing tasks and inputting data directly into the BPM engine. This was meant to acclimate the subjects with the technology.

Figure 7 depicts the process template for use case 2. In this scenario, an additional part (PCI card) has to be installed into a PC, analogous to adding or replacing a machine part in an

industrial scenario. As with use case 1, the subjects were able to directly interact with the process through their Magic Leap 1 (Figure 7a) and its pointing device, which could be hung on a belt if not required.

The following explains the separate workflow steps of use case 2 in detail:

- 1) "ask for assignment" script task: the BPM-Engine automatically triggers the IAC via the ARPFs ACI, at instance start, to determine the optimal worker for the process instance and assigns the selected worker in the BPMS (in this case the test subject)
- 2) The subject is notified of the new assignment in the AR App, via a red dot at the menu in the top right corner
- 3) "Go to PC" [AR1 task]: red spheres (anchors) are used to guide the subject to the destination (Figure 7b)
- 4) "Checklist" [AR2 task]: displays a pre modification checklist (Figure 7c), disconnecting power and other safety measures.
- 5) "Add PCI Card" [AR3 task]: AR-Video-Overlay is shown on how to open the PC (Figure 7e) and how to install the PCI card (Figure 7d). Afterwards the PC gets reconnected to power and restarts.
- 6) "Wait" [AR4 task]: subject awaits the startup and automatic system check (Figure 7g). This was mocked for the evaluation.
- 7) "Checklist" [AR5 task]: if an error was detected the subject was provided with a trouble shooting checklist (Figure 7f)
- 8) "Checklist" [AR6 task]: if no errors occurred the completion checklist (Figure 7h) is shown.

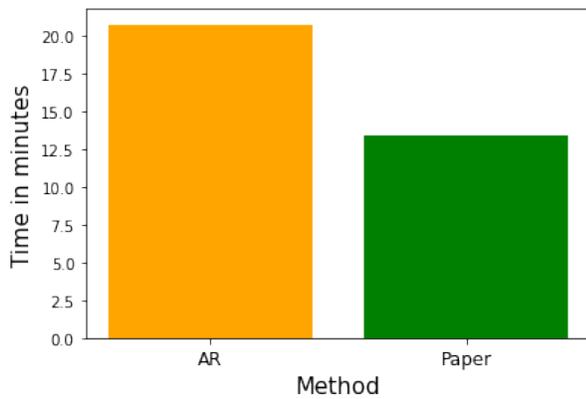


Figure 8. Task processing speed both evaluations combined.

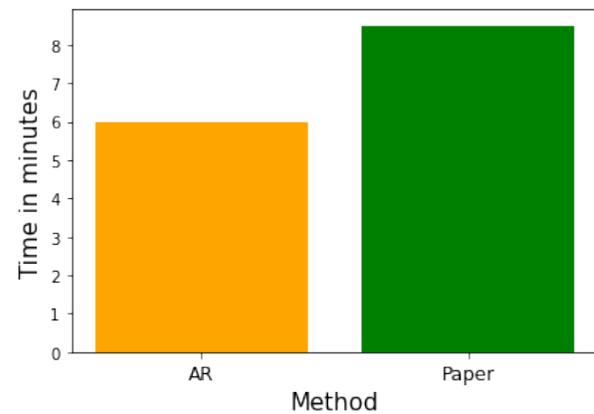


Figure 9. Task processing speed AR5 task in evaluation 2.

All subjects were able to complete the use cases in AR and with paper instructions, the starting order (AR first vs paper first) made no significant difference in the working speed. Overall, the conduction of all tasks (both use cases combined) with AR took around 50% longer than the completion of all tasks using the paper method as depicted in Figure 8. However, the handling of the second checklist ([AR 5]), was executed faster in AR than the paper version, cf. Figure 9. Further the subjects tended to forget check off points on paper more often than in the AR app, heavily reducing the error rate from 31% on paper to 10% using AR.

Most of the problems with the AR app further originated from problems with the precise anchoring of the overlays and the usage of the pointing device (which had to be put down and picked up again) rather than gesture control.

Overall, the AR approach proved slower for the people with no/little AR experience and requires further work, but people recognize its potential usefulness. This checks up with other AR projects, e.g., WART [15].

C. Findings and Discussion

While both evaluations resulted in positive findings, ARPF clearly isn't suitable for every use case nor user group. Although the framework itself provides powerful and efficient assignment algorithms and can be tweaked to the specific need of the BPM setup, some setup is still required. This includes specifying the requirements and dangers for each task, adding the additionally required agent data (e.g., their qualification), etc. The more complex the processes are and the more agents exist in the system the more setup time is involved. However, the value of ARPF increases as more agents are available for tasks, as it can provide error-less exact assignments for a large set of agents. The initial setup process can further be accelerated by using templates for tasks, with default requirements already specified, and only manually tweaking special tasks with hard constraints, such as hazards or hard requirements regarding agent qualification.

Furthermore, the evaluation has shown that the AR support can greatly reduce the error rate and accelerate the execution of suitable tasks. However, the AR support can also lead to

slowdowns of tasks if human agents are not familiar with AR devices or the tasks are not suited for AR. This makes ARPF vulnerable for the golden hammer anti-pattern [16]. Therefore, a thoughtful selection has to take place where to use AR support and where to rely on the traditional methods.

Overall the evaluation shows that ARPF can be an powerful tool and extension to classical BPMSs when applied correctly. Small companies most likely would not profit from the assignment component, as the initial setup would be cost prohibitive. Medium/large companies, however, could greatly benefit from the more sophisticated assignment process. The AR support has to be applied with care to suitable tasks, but used correctly could be beneficial for companies no matter their size.

V. RELATED WORK

Generalized context models are difficult to achieve and are not prevalent, as a survey on context models conclude [17]. An example is presented in [18]. The model is heavily tailored towards general pervasive computing scenarios and lacks several components crucial for Industry 4.0 AR processes. In contrast, the ARPF context model presented is rather specific and yet readily extensible, due to its three-layer context based on global, process, and activity context. Furthermore, the integration of context into process languages is challenging because they are not flexible enough, as stated in [19]. The contribution also proposes a BPMN extension for context integration, which is, in turn, tailored heavily towards mobile processes and not suitable for Industry 4.0 production.

Focused on context processing the Java Context Aware Framework [20] is a technical object- and service-oriented framework targeting modeling context changes via rules. However, the processing of such rules is forwarded to the application layer. In JCOOLS [21], this limitation is overcome by integrating JCAF with the Drools rule engine. The approach taken is rather complicated and generic, lacking support for both programmers and end users.

Examples of context modeling approaches include Coutaz and Crowley [22] and Ghiani, Manca, and Paternò [23]. However, these approaches primarily target the creation of context

rules by the application developer that can later be completed with concrete values by end users, without providing the execution infrastructure.

There are also contextual approaches for Industry 4.0 production. Giustozzi et al. [24] provide a context model for industry 4.0 processes. Some of the mentioned entities are similar to the ones in the ARPF. However, the model is ontology-based and the paper primarily deals with logical relations of the concepts, which makes concrete implementation in an industry-ready system problematic. Furthermore, only a model is presented, lacking other components for integration process enactment. BPMN4CPS [25] combines BPMN with CPS to add resources and context data to a business process for increased automation, but it does not integrate AR directly. Another model for Industry 4.0 production based on ML is presented in [26]. This model, however, is also not applicable for enactment of AR processes, as it primarily deals with predicting the degradation of the state of machines.

Another approach is taken by Tasdemir and Toklu [27]: it focusses on fuzzy task assignment and integrates BPM concepts. The described system is not suitable for the Industry 4.0 scenario, as it focuses on teams and the social relationships of the worker in the team. In addition, it lacks other components such as a real-time data context model.

Work related to the combination of context with AR tasks includes Blattgerste et al. [28], where AR glasses provide mobile assistive instructions. However, it was largely restricted to one concrete scenario rather than a generic business process. In BPMN4SGA [29] BPMN is extended for Smart Glasses, but primarily for documentation purposes rather than actionable AR content. In contrast, in our approach AR Actions are modeled and implemented via predefined AR templates containing attributes covering nearly all BPMN elements. Our AR application interprets the templates and sends feedback to the BPMN modeling application, avoiding the necessity of implementing or syncing steps with the BPM engine. SenSoMod [30] adds context-awareness to conventional non-production applications such as email, calendar, etc. Gronau & Grum [31] combined the Knowledge Modeling and Description Language (KMDL) with AR, projecting sensor data and process step association onto the visible machines, yet it lacks concrete tailored task guidance. HoloFlows [32] is an AR process modeling approach for the Internet of Things (IoT), utilizing a simple state-machine and custom notation that lacks BPMN support and integration with mature BPMS - vital for production settings.

In summary, ARPF provides a unique approach for contextual processing for Industry 4.0 processes with human AR tasks, supporting integration with existing BPMS and utilizing a BPMN extension to include AR and context in new and existing process models. Other approaches lack the inclusion of information needed for representing processes and their connection to AR devices and workers, machines, and resources with their specific contextual properties and rules. In addition, most of these approaches do not present an integrated framework for comprehensively supporting process enactment

in such complicated domains utilizing real-time data.

VI. CONCLUSION

This contribution described our ARPF approach for incorporating contextual factors crucial for AR tasks into Industry 4.0 production processes. The presented framework incorporates components to simplify the integration of such factors when modeling the processes and utilizes live data from different sources while executing them. This enables context-aware process enactment, which can improve process quality capabilities such as optimal task resource assignments, improved cost efficiency, and better support for AR user activities. Furthermore, by providing bi-directional communication interfaces between the process and the AR task, the latter can be seamlessly integrated into the process.

We further implemented a prototype integrating our approach with two prevalent BPMS. The prototype shows that the integration with real BPMS is feasible and achievable with little effort. Further, we conducted an evaluation executing a comprehensive simulation scenario with our prototype. Our findings suggest that our approach can lead to various improvements for Industry 4.0 processes with AR tasks. Task assignments can be improved by incorporating contextual factors. Further, AR task execution can be better supported with matching contextual information. The empirical part of our evaluation focused on AR tasks and showed slower execution times but better accuracy and lower error rates with AR support. However, this evaluation also suggests potential because the users were not used to AR devices. We thus expect faster execution times to be observed in daily usage. Thus, the overall process execution can be improved, resulting in better resource usage and cost savings. Moreover, other factors, such as worker safety can also be taken into account and be seamlessly integrated into the processes.

Future work includes: the optimization of our context-integrated process editor to improve its appearance and usability; integration of ARPF with further BPMS; application of ARPF to other domains; further improvements to the BPMN 2.0 extension; and a comprehensive empirical evaluation in a real production environment.

ACKNOWLEDGMENTS

The authors wish to acknowledge Felix Gräber's contribution regarding the technical foundation of the IAC. This work was partially funded via the PARADIGMA project by "Zentrales Innovationsprogramm Mittelstand" (i.e., the "Central Innovation Programme for small and Medium-Sized Enterprises (SMEs)"), of the Federal Ministry for Economic Affairs and Energy of the Federal Republic of Germany.

REFERENCES

- [1] G. Grambow, D. Hieber, R. Oberhauser, and C. Pogolski, "Supporting augmented reality industry 4.0 processes with context-aware processing and situational knowledge," in *eKNOW 2021*, 07 2021, pp. 29–36.
- [2] R. Baheti and H. Gill, "Cyber-physical systems," *The impact of control technology*, vol. 12, no. 1, pp. 161–166, 2011.
- [3] T. Allweyer, *BPMN 2.0: introduction to the standard for business process modeling*. BoD–Books on Demand, 2016.

- [4] G. Grambow, R. Oberhauser, and M. Reichert, "Contextual injection of quality measures into software engineering processes," *Int'l Journal on Advances in Software*, vol. 4, no. 1&2, pp. 76–99, 2011.
- [5] G. Grambow and R. Oberhauser, "Towards automated context-aware software quality management," in *2010 Fifth International Conference on Software Engineering Advances*. IEEE, 2010, pp. 347–352.
- [6] G. Grambow, R. Oberhauser, and M. Reichert, "Towards automatic process-aware coordination in collaborative software engineering," in *6th Int'l Conference on Software and Data Technologies*. ScitePress, 2011, pp. 5–14.
- [7] *Business Process Model and Notation(BPMN) - Version 2.0.2*. Object Management Group, 12 2013, retrieved: 2021.06.10. [Online]. Available: <https://www.omg.org/spec/BPMN/2.0.2/PDF>
- [8] G. Grambow, D. Hieber, R. Oberhauser, and C. Pogolski, "A context and augmented reality bpmn and bpmx extension for industrial internet of things processes," in *9th International Conference on Business Process Management (BPM 2021) Workshop Proceedings*. Springer, 2021.
- [9] Camunda. Last Visited: 2021.06.10. [Online]. Available: <https://camunda.com>
- [10] L. A. Zadeh, "Fuzzy logic," *Computer*, vol. 21, no. 4, pp. 83–93, 1988.
- [11] G. Grambow, D. Hieber, and R. Oberhauser, "Intelligent task assignment in industry 4.0 production processes utilizing fuzzy sets," in *INTELLI 2021, The Tenth International Conference on Intelligent Systems and Applications*, 07 2021, pp. 1–9.
- [12] M. Reichert, "Enabling flexible and robust business process automation for the agile enterprise," in *The Essence of Software Engineering*. Springer, Cham, 2018, pp. 203–220.
- [13] G. C. Hillar, *MQTT Essentials-A lightweight IoT protocol*. Packt Publishing Ltd, 2017.
- [14] M. Proctor, "Drools: A rule engine for complex event processing," in *Applications of Graph Transformations with Industrial Relevance*. Berlin, Heidelberg: Springer Berlin Heidelberg, 2012, pp. 2–2.
- [15] D. Haniff and C. Baber, "User evaluation of augmented reality systems," in *Proceedings on Seventh International Conference on Information Visualization, 2003. IV 2003.*, 2003, pp. 505–511.
- [16] A. H. Maslow, *The Psychology of Science - A Reconnaissance*. Harper & Row, 1966.
- [17] C. Bolchini, C. A. Curino, E. Quintarelli, F. A. Schreiber, and L. Tanca, "A data-oriented survey of context models," *SIGMOD Rec.*, vol. 36, no. 4, p. 19–26, Dec. 2007.
- [18] K. Henriksen, J. Indulska, and A. Rakotonirainy, "Modeling context information in pervasive computing systems," in *Pervasive Computing*. Berlin, Heidelberg: Springer Berlin Heidelberg, 2002, pp. 167–180.
- [19] J. Dörndorfer and C. Seel, "A meta model based extension of bpmn 2.0 for mobile context sensitive business processes and applications," *Proc WI 2017*, pp. 301–315, 2017.
- [20] J. Bardram, "The java context awareness framework (jcac) – a service infrastructure and programming framework for context-aware applications," vol. 3468, 05 2005, pp. 98–115.
- [21] J. Park, H.-C. Lee, and M.-J. Lee, "Jcools: A toolkit for generating context-aware applications with jcac and drools," *Journal of Systems Architecture*, vol. 59, no. 9, pp. 759–766, 2013.
- [22] J. Coutaz and J. L. Crowley, "A first-person experience with end-user development for smart homes," *IEEE Pervasive Computing*, vol. 15, no. 2, pp. 26–39, 2016.
- [23] G. Ghiani, M. Manca, and F. Paternò, "Authoring context-dependent cross-device user interfaces based on trigger/action rules," *Proceedings of the 14th International Conference on Mobile and Ubiquitous Multimedia*, pp. 313–322, 2015.
- [24] F. Giustozzi, J. Saunier, and C. Zanni-Merk, "Context modeling for industry 4.0: an ontology-based proposal," *Procedia Computer Science*, vol. 126, pp. 675–684, 2018.
- [25] I. Graja, S. Kallel, N. Guermouche, and A. H. Kacem, "Bpmn4cps: A bpmn extension for modeling cyber-physical systems," in *2016 IEEE 25th International Conference on Enabling Technologies: Infrastructure for Collaborative Enterprises (WETICE)*, 2016, pp. 152–157.
- [26] J.-R. Ruiz-Sarmiento *et al.*, "A predictive model for the maintenance of industrial machinery in the context of industry 4.0," *Engineering Applications of Artificial Intelligence*, vol. 87, p. 103289, 2020.
- [27] C. Tasdemir and C. Toklu, "Qos driven dynamic task assignment for bpm systems using fuzzy logic," in *Emerging trends and challenges in information technology management*. Hershey, Pennsylvania (701 E. Chocolate Avenue, Hershey, Pa., 17033, USA: IGI Global, 2006.
- [28] J. Blattgerste, B. Strenge, P. Renner, T. Pfeiffer, and K. Essig, "Comparing conventional and augmented reality instructions for manual assembly tasks," in *Proceedings of the 10th International Conference on Pervasive Technologies Related to Assistive Environments*, ser. PETRA '17. New York, NY, USA: Association for Computing Machinery, 2017, pp. 75–82.
- [29] J. Vogel and O. Thomas, "Generating smart glasses-based information systems with bpmn4sga: A bpmn extension for smart glasses applications," in *Wirtschaftsinformatik*, 02 2019.
- [30] J. Dörndorfer, C. Seel, and D. Hilpoltsteiner, "Sensomod-a modeling language for context-aware mobile applications," in *Multikonferenz Wirtschaftsinformatik (MKWI), 1435-1446*, 03 2018.
- [31] M. Grum and N. Gronau, "Integration of augmented reality technologies in process modeling - the augmentation of real world scenarios with the kmml," 7 2017, pp. 206–215.
- [32] R. Seiger, M. Gohlke, and U. Abmann, "Augmented reality-based process modelling for the internet of things with HoloFlows," in *Enterprise, Business-Process and Information Systems Modeling*. Springer International Publishing, 2019, pp. 115–129.

A Reference Ontology for Collision Avoidance Systems and Accountability Integrated with DAIDALUS

David Martín-Lammerding

Department of Stats. Comput. Sci. Math
Public University of Navarre (UPNA)
Pamplona, Spain
email:david.martin@unavarra.es

José Javier Astrain

Department of Stats. Comput. Sci. Math
Public University of Navarre (UPNA)
Pamplona, Spain
email:jozej.astrain@unavarra.es

Alberto Córdoba

Department of Stats. Comput. Sci. Math
Public University of Navarre (UPNA)
Pamplona, Spain
email:alberto.cordoba@unavarra.es

Abstract—Unmanned Aerial Systems (UASs) will be deployed massively in urban areas to implement logistic and security applications, with lower cost and more flexibility than manned aircraft. An increasing number of UAS missions requires an improvement of their safety by equipping UASs with Collision Avoidance Systems (CASs). CAS implementations require data from the UAS environment to identify conflicts and to perform an avoidance maneuver if required. UAS generates heterogeneous data from multiple sources, like the Flight Control Unit (FCU), the Global Navigation Satellite System (GNSS), a radio receiver, an onboard-camera, etc. However, each CAS implementation represents, processes and stores conflict data in a different way. Therefore, there is a lack of standards that simplify their development and homologation. To solve this situation, we present a reference knowledge model for any CAS for UAS implemented as a novel application ontology, called *Dronetology-cas*, and its integration with DAIDALUS, a CAS developed by NASA. *Dronetology-cas* provides a unified semantic representation within an ontology-based triplet store designed to run in a Single Board Computer (SBC). Its semantic model provides advantages, such as interoperability between systems, machine-processable data and the ability to infer new knowledge. It is implemented using semantic web standards, which contribute to simplify an operational safety audit. Additionally, we integrate *Dronetology-cas* with DAIDALUS and verify it with two scenarios where conflict data from external sources are considered to improve UAS safety.

Index Terms—*Semantic reasoning; ontology; DAIDALUS; UAS; knowledge; conflicts; anti-collision; sensor; embedded; air traffic.*

I. INTRODUCTION

The use of Unmanned Aerial Systems (UASs) improves efficiency in logistics applications, infrastructure inspection, emergency situations, etc. as it avoids pilot risk. However, their flights are limited to certain areas of the airspace to avoid encountering other aircrafts. Therefore, each UAS must be equipped with new safety systems to fly safely in a shared airspace, like Collision Avoidance Systems (CASs). CASs detect aircraft, discover potential collision hazards and decide maneuvers to avoid collisions that may change the flight plan initially configured.

A massive use of UASs will imply an increasing collision risk that involves ensuring accountability for all UASs mis-

sions. The accountability principle requires UAS operators to take responsibility for what their UAS do in a mission and how they comply with traffic management authorities. UAS operators must have appropriate records to be able to demonstrate their compliance. The accountability of an UAS flight must be ensured because any incident or accident must be able to be investigated by surveyors or authorities. In the worst case, a collision may occur, which must be investigated to determine the cause and to improve CAS.

In our previous work [1] published at SEMAPRO, we present a novel ontology, *Dronetology-cas*, for CAS for UAS. From those results, we develop an integration with a reference CAS implementation, called DAIDALUS (Detect and Avoid Alerting Logic for Unmanned Systems [2]), as it implements the functional requirements specified in DO-365, the Minimum Operational Performance Standards (MOPS) for UAS developed by RTCA (Radio Technical Commission for Aeronautics) Special Committee 228 (SC-228). DAIDALUS avoidance does not depend on communications with centralized systems, as any delay in making a decision increases the risk of collision. Our based knowledge application also requires to be executed onboard to reduce communication latency. So, we deploy *Dronetology-cas* in a Single Board Computer (SBC), suitable for mounting on an UAS, to verify its capabilities of collision avoidance, collision investigation and if response time is reasonable. *Dronetology-cas* includes a Knowledge Base (KB), which consists of triplets of data collected by onboard sensors, external conflict data and inferred knowledge during the UAS mission. *Dronetology-cas* uses software components to interact with the CAS, onboard sensors and external data sources.

The investigation of aviation incidents and accidents is today recognized as a fundamentally important element of improving safety. The International Civil Aviation Organization (ICAO) releases Annex 13 [3] that requires various States to establish and maintain an accident and incident database to provide an effective analysis of information on actual or potential safety deficiencies. UASs traffic management will be probably inspired by actual commercial aviation recommendations so similar requirements are expected. However,

the increasing number of UASs and the autonomy required for their operation forces process standardization and automation. Ontologies unify schemes for exchange of information and provides access to stakeholders and provide a common format for identifying actual or potential safety deficiencies. Therefore, Dronetology-cas may be integrated with any CAS typology, whose design factors are depicted in Figure 1.

There are external air-traffic data that can be used in collision avoidance. As an example, we consider data from Notice To Airmen (NOTAM), provided by an aviation authority. It alerts aircraft pilots of potential hazards along a flight route or at a location that could affect the safety of the flight. We insert NOTAM data in the KB during a simulated UAS flight, when connectivity allows to download it from an external data source, to consider conflicts that may not be detected with onboard sensors.

ADS-B (Automatic Dependent Surveillance - Broadcast) [4] is a surveillance system that replaces the information currently obtained from radars. It allows to broadcast to other aircrafts the position, obtained from a Global Navigation Satellite System (GNSS), and other flight data. These signals are received by ground or onboard aircraft receivers. UASs can deploy an ADS-B transceiver as part of its integration into the common airspace. Therefore, we also consider ADS-B data in the KB during the simulation performed to improve situational awareness and minimize collisions.

The rest of the paper is structured as follows. Section II presents the state of the art of CAS and accountability systems, Section III defines the problem statement and Section IV describes our contribution. The ontology design is presented in Section V. Section VI is devoted to the integration architecture. Section VII formulates ontology Competency Questions (CQs) and Section VIII summarizes experimental simulations results. Section IX presents the conclusions and references end the paper.

II. RELATED WORK

We review ontology applications, autonomous driving and CAS implementations to identify different approaches. Among them, autonomous driving applications shares some requirements with autonomous UAS and CAS. CAS for autonomous driving requires an accurate localization of the car provided by multiple sensors, as described in [6]. This can also be applied to UASs localization, as in landing maneuvers that require precision, using our proposed ontology-driven system, as it can deal with any precision available. Therefore, a knowledge layer, like we propose, allows to apply it easily to other systems and vehicles.

Ontologies are commonly used to store knowledge in domains related with UAS and CAS, like sensors and air traffic. For the former, we review the Semantic Sensor Network (SSN) [7] ontology that can be used to model UAS as sensors, as it is one of the most widespread application of UAS is data gathering. However, the SSN ontology lacks concepts to model the UAS mission and the CAS, so an extension is required which implies a larger ontology. A large ontology increases

memory consumption and we want to run the system in a SBC with limited resources.

Regarding the air traffic domain, we review the propose ontology[8] that applies semantic technologies to air traffic in order to unify heterogeneous data from multiple sources. The ontology implementation presented is performed centralized. However, our proposal is a decentralized ontology implemented in a SBC mounted on each UAS to serve as a knowledge base for the CAS.

Ontology performance is another issue that we review as we want to improve CAS performance using a SBC with limited computing resources. [9] presents a *light-weight* ontology for embedded systems whose design reduces concepts, complexity and query times, compared to the SSN ontology. It is intended for the sensor domain and, therefore, it has limitations for modeling a CAS for UAS, as we want to achieve a knowledge repository to improve CAS performance. However, our proposal also limits the number of classes and the relations for just the necessary.

There are multiple CAS implementations for UAS, but we only consider ACAS-Xu [10] and DAIDALUS [2], as both CAS have their source code available and are two reference CAS implementations. Both CASs require a specific configuration for considering the same avoidance maneuvers and conflict scenario. Given the same scenario, their output formats are different as shown in [11]. A limitation of both CASs is that they do not share a common conceptual model that simplify the usage of knowledge, like the solution we propose.

Multi-Task Learning is an approach for autonomous driving applied to obtain more interpretable results from raw sensor data, thanks to human-readable intermediary representation, as presented in [12]. Our ontological approach can also be considered an intermediate representation, but we also want to facilitate its reusability in other systems, as one key feature of our proposed solution is to improve interoperability. Therefore, we consider that semantic technologies will achieve both objectives.

The accountability of an UAS flight must be ensured because any incident or accident should be able to be investigated by surveyors or authorities. There are systems similar to black boxes for UAS [13]–[15]. They store the UAS's route and the CAS's status. However, the decision-making process prior to a maneuver is complex and its recording is not provided in these systems, therefore, we improve the accountability of a CAS recording decisions and maneuvers in the KB with a common vocabulary that facilitates interoperability.

III. PROBLEM STATEMENT

The data required by a CAS depends on how the main design factors are combined. The main concepts of CAS used in the design of Dronetology-cas are described below.

A conflict between two UAS occurs when minimum separation between them is lost. Figure 2 shows a conflict between *local UAS* and *remote UAS* and the protection distance between them, d_p . A loss of separation does not always imply a future

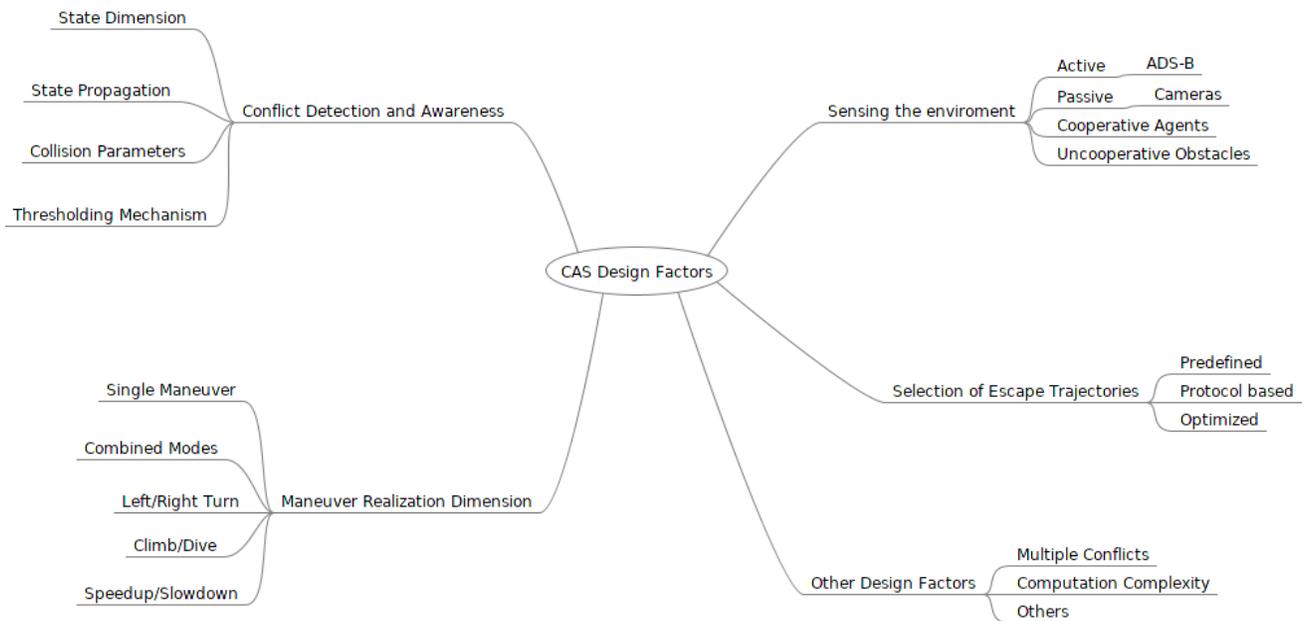


Fig. 1. CAS design factors. Based on [5].

collision, but it is a key safety indicator. A CAS deployed in an UAS allows to maintain a minimum safe separation between UASs. Once a conflict is detected, the onboard CAS deviates the UAS to a safer path. The number of simultaneous conflicts are denoted as N_C . Time to collision t_{c} is the time required to collide two UAS if both UASs continue at their current speed and on the same path. Lower t_{c} values correspond to higher risk of collision and it is considered to prioritize conflicts. Very Low Level airspace (VLL) is the space below 500 ft, measured Above Ground Level (AGL). It is the part of the airspace intended for most of the new UAS applications and it will concentrate the largest number of UAS conflicts.

CASs are based on different technologies that collect data from the UAS environment using sensors and/or collaborative elements based on radio receivers/transmitters. UAS can deploy collaborative elements and non-collaborative sensors. A collaborative element broadcast its position and bearing within its coverage and receives from other aircrafts. ADS-B is the most common standard applied in collaborative systems. A non-collaborative sensor detects obstacles and conflicts without requiring other to implement the same system. Technologies applied to non-collaborative systems are vision cameras [16], LIDAR [17], SONAR [18], Radar [19], etc. [20] presents a complete survey of the main technologies applied to sensors for conflict detection.

Most CASs for UASs are distributed, so they run in a SBC mounted on each UAS. However, the size of the UAS limits the weight of the payload, which limits the type and power of processor that can be used. Any software component used in a distributed CAS implementation should be non-compute-intensive to ensure a reasonable response time.

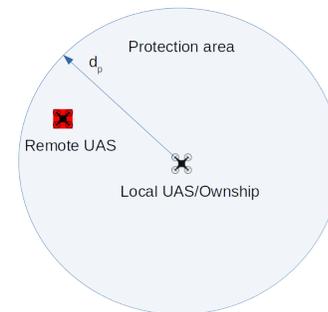


Fig. 2. Conflict between a local UAS (ownship) and a remote UAS.

IV. CONTRIBUTION

In this paper, we develop the integration of Dronetology-cas with DAIDALUS and verify the integration with two scenarios where knowledge provided by Dronetology-cas improves DAIDALUS collision avoidance. We also describe the ontology Dronetology-cas and its main characteristics. Dronetology-cas improves DAIDALUS as it minimizes the processing of unnecessary conflicts and increases the situational awareness of the UAS translating NOTAMs to conflicts.

Dronetology-cas provides key advantages over other repositories or log storage implementations, as it is ontology based and can be queried using SPARQL (SPARQL Protocol and RDF, Resource Description Framework, Query Language). Another Dronetology-cas key features that we verify with the simulations performed are reasonable response time, modifiability, ease of maintenance, built-in inference capabilities and potential for reuse.

V. DRONETOLOGY-CAS: THE APPLICATION ONTOLOGY

Dronetology-cas is an application ontology derived from the domain ontology Dronetology [21]. The domain of Dronetology is UASs. Dronetology-cas formal specification is based on the design factors of a CAS. Next, we review Dronetology before describing Dronetology-cas.

A. Dronetology: The domain ontology

The purpose of Dronetology is to describe concepts that define the components of any UAS, the missions it performs and the environment that surrounds it. Its main applications are the management of bill of materials, the improvement of flight efficiency and autonomous decision making. Dronetology imports external ontologies to avoid redefining concepts from other domains, like SSN for onboard UAS sensors. Another advantage of importing widespread ontologies is that available data from other applications can be integrated easily.

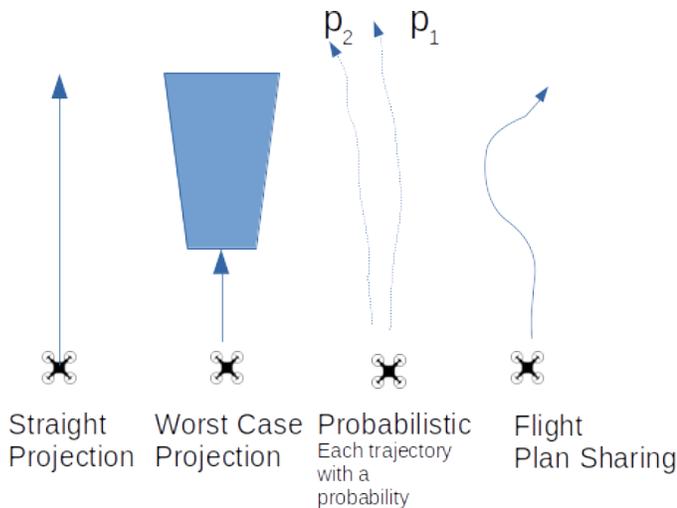


Fig. 3. Methods used for projecting current encounter's information. Based on [5].

B. Dronetology-cas description

We derive the Dronetology-cas application ontology from the Dronetology domain ontology adding concepts from the CAS domain. Its design use abstract concepts that can be used in any CAS integration. The ontology is accessible in [22].

Dronetology-cas allows auditing the CAS decision making process, as the KB stores the planned flight plan, the CAS status and the real flight path of the mission. Conflicts and their temporal variations are detected by onboard sensors and stored in the KB. Differences between planned and real flight can be retrieved using SPARQL queries, as flight plan changes usually are related to conflicts.

Knowledge is obtained from data collected from sensor systems and collaborative elements, and stored in the KB. Data sources are sensors, the Flight Control Unit (FCU) and the GNSS. Inference improves the CAS decisions thanks to

knowledge derived from the data. Dronetology-cas has an inference engine that generates new knowledge by applying semantic rules to the KB. The rules are expressed in SPARQL statements [23], [24]. Defined rules inference a conflict's attribute, an evasive trajectory method, a maneuver attribute, etc.

A common feature of a CAS is that it usually runs in a loop with an operation frequency in order to update the internal representation of conflicts, the situation awareness and the current maneuver. This is modeled in Dronetology-cas with the concept of *Iteration*. Dronetology-cas stores CAS status, UAS telemetry and conflicts for each *Iteration* to audit the system. Data collected from sensors are also related to the *Iteration* to provide a complete picture of the environment and the CAS. Dronetology-cas simplifies the integration of data from different sources. It integrates data from any sensor system by defining generic classes, which are not directly dependent on the technology and the implementation. These classes are *NoCollaborativeData* and *CollaborativeData* and both extend *InputData*.

Another common feature of a CAS is to estimate future positions of conflicts, called a projection, in order to obtain a maneuver that will avoid a future collision. As an example of the above, Dronetology-cas allows to choose the method to estimate the future position from multiple options available, as depicted in Figure 3. If the conflict has been detected only through a vision camera, the uncertainty about the heading of the conflict is higher, so the most appropriate method of estimating may be the *Worst Case* method. On the other hand, if the conflict has been detected by a collaborative element, the heading is known and there is less uncertainty. In this case, the *Straight Projection* method is the most appropriate. Both decisions can be derived easily using Dronetology-cas with a rule that relates the conflict data source with the most appropriate estimated method for the projection.

When the CAS decides a maneuver to avoid a collision, Dronetology-cas stores every UAS position and groups them with a individual of class *Maneuver*. Thus, Dronetology-cas relates multiple specific-maneuvers concepts, like *left-turn*, with a set of positions, which allows any combination of trajectories, altitudes and speeds. Similarly, the dynamics of conflicts in the 3D space are stored in Dronetology-cas as different positions at different times. On the contrary, full trajectory prediction made by the CAS are not stored in Dronetology-cas as it is a highly variable data.

C. Dronetology-cas design

The CAS design factors, presented in previous section, are used in the ontology design. Concepts defined in Dronetology-cas are abstractly modeled to fulfill any CAS requirement and simplify the integration. The relevant information that enables making a CAS accountable is identified and annotated with adequate ontology terms. Next, we review each design factor and how Dronetology-cas implements each one.

The first design factor considered in the design is the type of onboard sensors. Dronetology-cas models every onboard

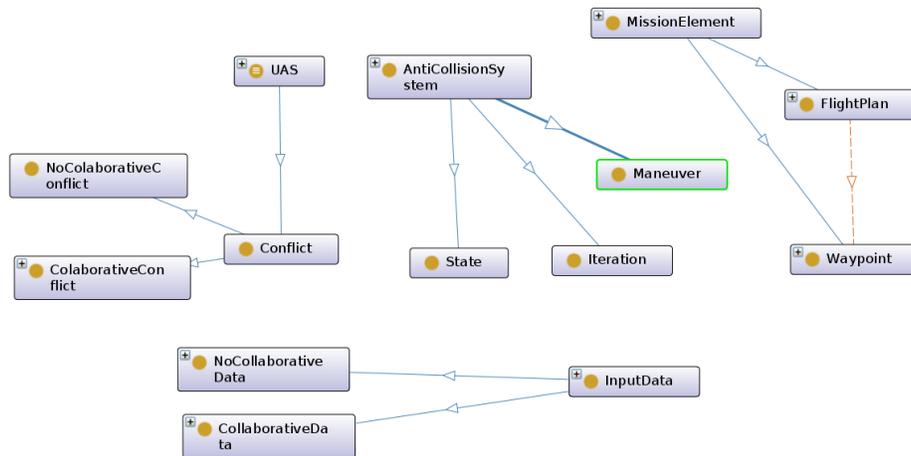


Fig. 4. Dronetology-cas main classes

sensor as an abstract data source so it reduce classes and simplifies the design, avoiding to define specific technologies and attributes for each sensor system. Therefore, sensors are classified between collaborative and non-collaborative.

Dronetology-cas stores conflict locations in every *Iteration* (defined by the period of the CAS) in order to store data for ensuring the system accountability. The ownship location is also periodically stored in Dronetology-cas, distinguishing planned location from locations derived from an evasive maneuvering obtained from the CAS.

The Dronetology-cas design implements knowledge inference to improves the CAS response. For example, inference can classify conflicts, aggregate data from multiple sensors or dismiss a conflict. Inference from historical data are also supported. For example, when a conflict's attribute are not available, like speed, it can be inferred from the conflict past locations. The method to calculate an evasive trajectory and the associated maneuver depends on the CAS implementation.

Dronetology-cas has been designed considering the computational limitations of SBC. Thus, memory usage has been reduced by limiting the number of classes in the model and avoiding importing auxiliary ontologies.

The main classes of Dronetology-cas are *UAS*, *MissionElement*, *InputData*, *AntiCollisionSystem* and *Conflict*. Figure 4 shows the main Dronetology-cas classes.

The class *UAS* describes unmanned aircraft including the communication systems and the ground base. The class *Conflict* is a subclass of *UAS* so in our model only *UAS* can be conflicts. *MissionElement* is a class that enclose all the elements of a mission. The classes *Waypoint* and *FlightPlan* derive from *MissionElement*.

The class *InputData* represents any data collected from a

sensor (non-collaborative), from a collaborative element (radio receiver), from the GNSS or from the FCU. The concepts *NoCollaborativeData* and *CollaborativeData* are derived from *InputData* to identify a conflict and its source type. The property *drone:detect* is an object property that relates individuals of *NoCollaborativeData* or *CollaborativeData* with individuals of class *Conflict*.

Some classes in Dronetology-cas have geographic data defined as datatype properties. The latitude and longitude are relative to the World Geodetic System 1984 (WGS84) coordinate system. The altitude is relative to Mean Sea-Level (MSL). To improve interoperability, the *Conflict* class uses *geo:wktLiteral* datatype with a WGS 84 geodetic latitude-longitude. This allows Dronetology-cas to implement a geospatial web service that could be reused and recombined to fulfill a user query using a common standard.

The class *AntiCollisionSystem* groups elements of any CAS. The classes *State*, *Maneuver*, *NextIterationLocation* and *Iteration* are derived from it. The state of the CAS are represented as instances of the class *State* with an attribute that codifies it. A class *Iteration* instance relates all the knowledge stored in the KB at an instant of time through the object-property *hasIteration*.

The class *Maneuver* defines a set of locations of the UAS when the CAS is active. CAS may calculate one or multiple location alternatives for the UAS to avoid the collision, stored as instances of the class *NextIterationLocalUASLocation*, grouped by an instance of the class *Maneuver* through the object-property *hasManeuver*. In every iteration, at least a new instance of *NextIterationLocalUASLocation* is stored in the KB and sent to the FCU by the CAS.

Flight safety compliance are implemented in Dronetology-

cas defining attributes of the class *Conflict* to translate the severity classification requirements defined in ESARR2 [25].

VI. SYSTEM ARCHITECTURE

Dronetology-cas integration with a CAS can be implemented in two ways: *repository-mode* or *knowledge-mode*. The *repository-mode* of Dronetology-cas is a data sink, intended for flight telemetry storage. The *knowledge-mode* extends the *repository-mode* adding a connector or an endpoint to provide knowledge. Figure 5 shows the system architecture for each integration mode.

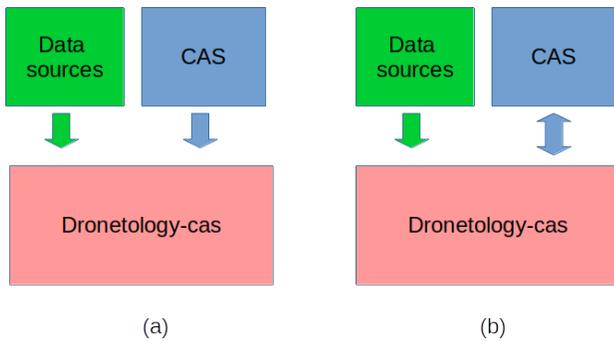


Fig. 5. Dronetology-cas integration alternatives: (a) *repository-mode* (b) *knowledge-mode*

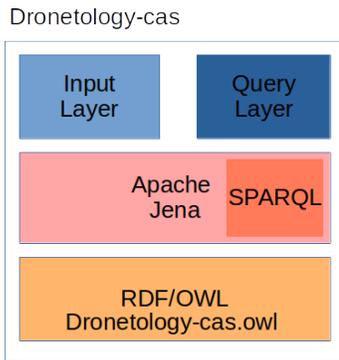


Fig. 6. Dronetology-cas system architecture.

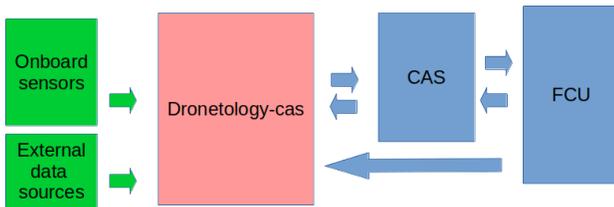


Fig. 7. Dronetology-cas *knowledge-mode* integration architecture detailed.

Dronetology-cas system architecture defines software components that allow its integration with other UAS subsystems, like a CAS, onboard sensor systems, the FCU and external data sources. The components of the Dronetology-cas architecture

consists of an input data layer and a query layer, as shown in Figure 6. The input layer, implemented for the *repository-mode*, is a software layer that insert into the KB collected sensor data, CAS status, GNSS locations, FCU telemetry and safety information form external providers.

The *knowledge-mode* architecture is detailed in Figure 7. The query layer, implemented for the *knowledge-mode*, is a SPARQL adapter that executes sentences and return values. The query layer provides a knowledge base criteria to select conflict detected or received by the onboard sensor. It adds implicit knowledge inference and reasoning capabilities to some CAS functions, such as conflict detection or new path selection. It can be configured as a SPARQL endpoint if HTTP interconnection is required for incident investigation, as federated SPARQL may be convenient to access other endpoints.

External conflict data sources relies on available connectivity to insert data in the KB. NOTAM providers and Ground Surveillance Radars (GSRs) are examples of external data sources.

Dronetology-cas is developed using the Web Ontology Language (OWL) language [26]. The main languages used to develop CAS (C, C++, Python) have implementations to process RDF triplets [27] and ontologies in OWL format.

Dronetology-cas architecture requires multi-platform compatibility to run in different hardware. Different software components are available to implement a semantic web stack, and among them, we select Apache Jena as it is the most common stack. Apache Jena is an open source Java framework for building Semantic Web and Linked Data applications. Additionally, its only dependencies are the Java Runtime Environment (JRE) so it can be deployed in every SBC that has a JRE implementation.

TABLE I
DRONETOLOGY-CAS COMPETENCY QUESTIONS

CQ ₁	How many conflicts are detected?
CQ ₂	Which UAS has the highest priority among the UAS in conflict?
CQ ₃	Which conflict has the shortest time to collision?
CQ ₄	Has the number of conflicts increased or decreased?
CQ ₅	How has been detected the conflict with a given UAS?
CQ ₆	How long it has taken to resolve a conflict?
CQ ₇	Has the distance flown been increased with respect to the flight plan?
CQ ₈	In which locations have there been conflicts?
CQ ₉	Where and when was the collision?
CQ ₁₀	How many UAS were in conflict before the collision?
CQ ₁₁	What UAS has it collided with?
CQ ₁₂	What maneuver was the UAS performing before the collision?

VII. COMPETENCY QUESTIONS

We define a set of Competency questions (CQs) to define the knowledge that has to be entailed in Dronetology-cas. These questions, listed in Table I, has been used to validate Dronetology-cas. Some CQs are suitable for an UAS mission audit process. Others can assist the CAS in a decision making process, when Dronetology-cas is integrated in *knowledge-mode*. There are CQs that are intended to find out how the conflict has been resolved, e.g., CQ₆, CQ₇ and CQ₈. Some

CQs help to find out what happened and how when a collision happens, e.g., CQ₉, CQ₁₀, CQ₁₁ and CQ₁₂.

In a *knowledge-mode* integration, the CAS uses the results of some CQs to make decisions. Continuing the previous example, the CQ *What type of conflict is X?* allows the CAS to select the most appropriate way of calculating the future position of the conflict. Other CQs are intended for a security audit of the CAS. An example of this is the CQ used to check when and where a collision occurred.

VIII. SIMULATION

Dronetology-cas verification is performed using simulations to assess its response time and its integration with DAIDALUS. Testing CAS with live-fly field experiments are costly and highly time consuming. Therefore, simulations are more convenient than to their flexibility and easy of configuration.

The first simulation is a performance evaluation of the KB of Dronetology-cas when queried with some selected CQs translated to SPARQL sentences. We develop a *simulated-CAS* that loads data in the KB during the simulation time to test the effect of the growth of the number of triplets stored in the KB and to measure the performance when executing SPARQL queries.

The second simulation performed is a Dronetology-cas *knowledge-mode* integration with DAIDALUS. A custom FCU simulator is developed in Java 8 to generate UAS flight positions (expressed in WGS-84 coordinates) from an initial flight plan. Multiple UASs can be simulated by the FCU simulator as it can generate positions of the ownship and the conflicting UASs sharing a common time reference.

We consider for the simulations the well known SBC Raspberry Pi 3 Model B+ (Pi3) [28], as it is shipped in large numbers and it has a huge user base. Its processor is a 1.2 GHz 64-bit quad core ARM Cortex-A53. The operative system installed to perform the simulations is the Raspberry Pi OS 32 bits. The FCU simulator runs in an external desktop computer that is connected with the Pi3 through a network connection.

A. Performance evaluation

Dronetology-cas performance is tested in both modes, *repository-mode* and *knowledge-mode*. The performance evaluation is executed using the *simulated-CAS* that loads the KB dynamically with data from conflicts and queries the KB using CQs translated to SPARQL sentences.

The most generic CQs have been selected to measure response times and memory footprint, as they are the most likely to be used in any integration mode. CQ₁ and CQ₃ are necessary for any auditing process to review conflicts and their status. CQ₅ and CQ₆ provide knowledge that the CAS can use to modify its response to conflicts. One hundred repetitions of each case were performed to calculate the mean and the standard deviation. The results obtained from the response times and memory footprint are shown in Table II. CQs considered are translated to SPARQL, available in [29].

Response time and memory footprint are measured with different number of triplets stored in the KB. Memory footprint

has been measured using the Java 8 API. The number of triplets with conflicts and CAS data grows as the UAS flies. Therefore, the flight duration determines the number of triplets stored in the KB. In our tests, we have simulated up to 10000 triplets that corresponds to 15 minutes of flight by inserting an average of 10 triplets per second.

The response time affects the CAS depending on the integration type chosen. In *repository-mode*, there are no strict response time requirements as it is not required a short response time. However, in *knowledge-mode*, the response time delays the CAS decisions. For our purpose, a suitable response time should allow to take a decision with the most recent data, before new data is available, that is, the response time should be below the refreshing rate of incoming data. Each sensor system has its refreshing rate ranging from 1 Hz of ADS-B until 20 Hz of a vision camera [30]. The response time of CQ₅ and CQ₆ obtained complies with the previous criteria as long as the number of triplets are below approximately 1000 triplets.

Figure 8 shows that Dronetology-cas response time increases when the number of triplets increases. Memory consumption grows as the UAS flies as well. That is, the duration of the UAS flight increases the response time. The worst response time is at the end of a flight. This result is due to our limited implementation of the software components that instantiates and queries the KB. An option to scale up is to have two instances of Dronetology-cas model, each with a different purpose, one instance for the *repository-mode* and the other for the *knowledge-mode*. The instance for the *repository-mode* should store all triplets, but the instance for the *knowledge-mode* should keep only recent triplets needed for the inference process.

B. Dronetology-cas knowledge-mode integration with DAIDALUS

DAIDALUS is a reference implementation of the algorithm for the Phase 1 Minimum Operational Performance Standards (MOPS) for Detect and Avoid (DAA). It is available a prototype implementations written in Java. It serves as a reference of a MOPS-compliant DAA algorithm so we select it to verify Dronetology-cas *knowledge-mode* integration.

Dronetology-cas integration with DAIDALUS v2.0.2b is developed in Java, using a knowledge layer that provides data location from the ownship and the remote conflicting aircraft to DAIDALUS.

The integrated system is simulated using the FCU simulator to generate the ownship location and other conflicting aircrafts when necessary. The simulation is a continuous execution that inserts locations in the KB while flying a defined flight plan. In each iteration, Dronetology-cas provides DAIDALUS the current position of the conflicting-UAS and the ownship. DAIDALUS calculates a response in form of a new location for the ownship that it is inserted in the KB. The simulations performed are autonomous flights where the ownship follows the corrective guidance provided by DAIDALUS without pilot interaction.

TABLE II
 RESPONSE TIME (IN MILLISECONDS) AND MEMORY FOOTPRINT (IN KILOBYTES) OF *repository-mode* AND *knowledge-mode* IMPLEMENTED IN APACHE JENA RUNNING IN A PI3.

No triplets	<i>Repository-mode</i>								<i>Knowledge-mode</i>							
	CQ ₁				CQ ₃				CQ ₅				CQ ₆			
	Response time		Memory footprint		Response time		Memory footprint		Response time		Memory footprint		Response time		Memory footprint	
	mean	sdev	mean	sdev	mean	sdev	mean	sdev	mean	sdev	mean	sdev	mean	sdev	mean	sdev
100	18.95	4.13	5025.28	1418.13	26.68	7.83	5101.67	1420.78	19.98	5.91	5104.45	1420.83	18.21	7.45	5100.81	1421.06
250	23.54	2.26	5200.98	1308.69	23.38	2.17	5241.25	1308.82	24.12	4.18	5211.77	1308.21	24.03	2.23	5233.40	1305.41
500	38.10	2.97	5441.26	1327.53	38.14	3.10	5441.51	1322.02	52.37	18.03	5377.53	1327.33	39.05	3.56	5491.02	1322.82
1000	74.42	17.78	5986.61	1332.65	67.47	3.10	5983.91	1336.00	68.49	3.18	6061.40	1331.63	68.74	3.10	5969.27	1319.01
2500	165.62	36.42	6997.64	1342.78	171.69	46.86	3362.77	1727.86	173.87	55.31	3875.86	1768.38	160.60	6.81	6972.95	1315.36
5000	320.98	64.36	6004.34	1718.44	320.30	63.49	5391.63	1587.92	355.39	100.29	5309.72	1510.33	324.63	65.44	5935.41	1724.54
10000	662.00	162.01	9545.00	1744.46	662.24	164.92	8440.54	2154.39	654.65	151.11	8560.70	2140.84	670.81	162.15	9595.25	1752.67

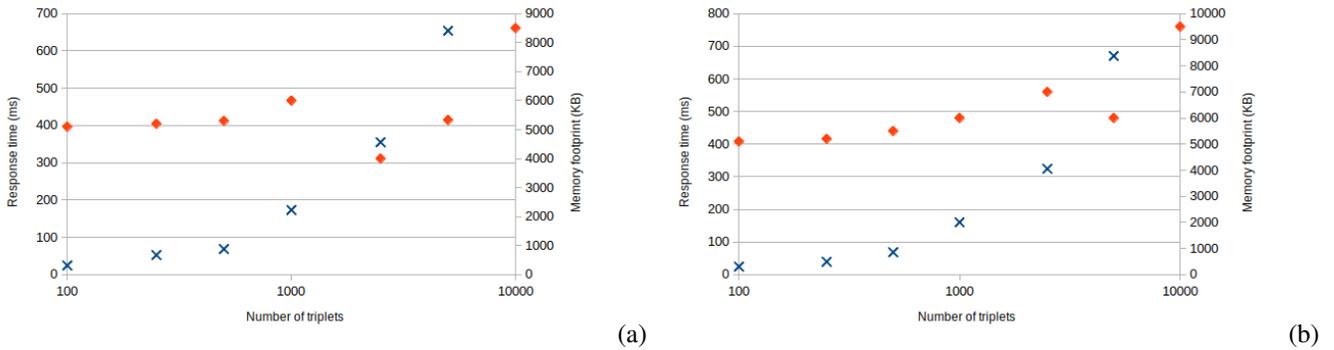


Fig. 8. Response time (x) and memory footprint (♦) for *knowledge-mode* for CQ₅(a) and CQ₆(b).

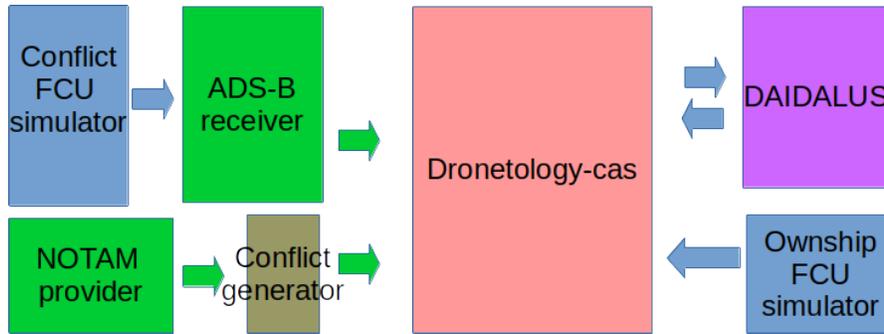


Fig. 9. Simulation architecture implemented to verify Dronetology-cas *knowledge mode* integration.

Next, we present two simulated scenarios for the *knowledge-mode* integration where a modification of the architecture depicted in Figure 9 is developed, whose main difference is that the KB is connected to the onboard sensors. The simulation of both scenarios are performed in a Pi3.

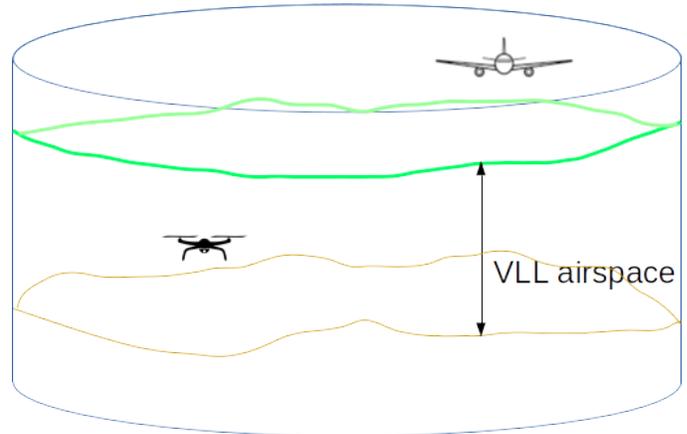


Fig. 10. Conflict not considered by Dronetology-cas as the flying airspace class of the aircraft and the UAS are not the same. Based on [31]

1) *Scenario 1: Using ADSB knowledge:* ADS-B is a common surveillance technology that increases situational awareness between both manned and unmanned operations. The availability of ADS-B equipment and its decreasing cost have contributed to mount it in UASs. Nowadays, the majority of aircraft broadcast ADS-B messages constantly. Starting from the year 2020, civil aviation aircraft in Europe and United States are required to be ADS-B compliant. Airspace is divided in classes specified by the International Civil Aviation Organization (ICAO) [32]. In our simulated scenario, depicted in Figure 10, manned aircraft and UASs do not share same airspace class. The developed FCU simulator inserts ADS-B data in the KB, as *ADBSConflict* class instances, that refers to a remote conflicting aircraft. Altitude is an attribute of the *ADBSConflict* class. It is obtained from the aircraft airborne position ADS-B message that is used to broadcast the position and altitude of the aircraft. It has the Type Code 9–18 and 20–22. When Type Code is from 9 to 18, the encoded altitude represents the barometric altitude of the aircraft. When the Type Code is from 20 to 22, the encoded altitude contains the GNSS altitude of the aircraft.

In our proposed scenario, a small UAS, equipped with an ADS-B transceiver, flies in VLL and receives ADS-B messages from an aircraft flying in the class A airspace. In a common configuration, every ADS-B message received are transformed into a conflict that are processed by DAIDALUS. However, DAIDALUS does not provide an evasive maneuver in this case because vertical separation is larger than the minimum configured, but this processing requires some time. When the number of conflicts flying in a different airspace class increases the delay may affect response time of the overall system. To see this effect we simulate multiple conflicts and the response time of DAIDALUS. Results, that are depicted in Table III, show that DAIDALUS Response Time (RT) increases when the number of conflicts increases. The mean of the RT has been measured repeating DAIDALUS execution 100 times for each number of conflicts.

TABLE III
DAIDALUS RESPONSE TIME CHANGING THE NUMBER OF
SIMULTANEOUS CONFLICTS.

Number of conflicts	0	1	2	3	4
DAIDALUS RT (ms)	4	157	175	174	185

Dronetology-cas integration in *knowledge-mode* avoids this delay due to conflicts flying in different airspace classes, as the criteria defined with a SPARQL sentence provides DAIDALUS only conflicts in the same airspace zone, avoiding aircraft that fly out of VLL airspace. The SPARQL sentence, depicted in Listing 1, defines a criteria that selects conflicts received by an ADS-B transceiver, that fly in VLL (below 500 ft) and its ADS-B type is UAV. When repeating the previous simulation with the Dronetology-cas integration, no conflicts are returned to DAIDALUS.

```
PREFIX rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns#>
```

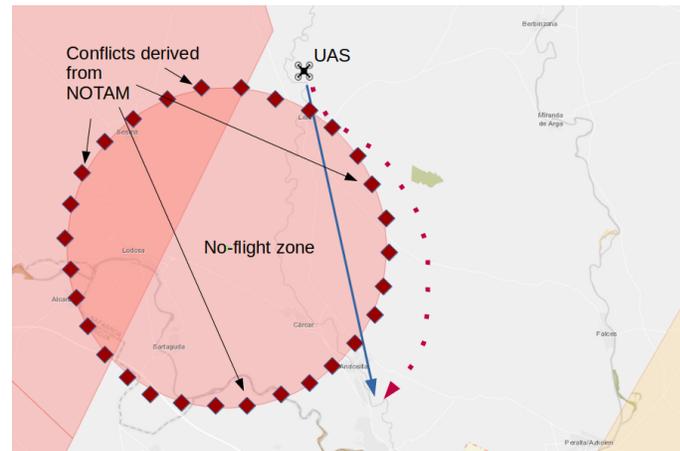


Fig. 11. A no-flight zone defined in a NOTAM is translated to multiple conflicts in the no-flight zone boundary. UAS planned flight (blue) and evasive path (red) are represented.

```
PREFIX owl: <http://www.w3.org/2002/07/owl#>
PREFIX rdfs: <http://www.w3.org/2000/01/rdf-schema#>
PREFIX xsd: <http://www.w3.org/2001/XMLSchema#>
PREFIX drone: <http://www.dronetology.net/dronetology-cas.owl#>
```

```
SELECT ?latitude ?longitude ?altitude
WHERE
{
    ?conflict a drone:Conflict.
    ?conflict drone:hasLatitude ?latitude.
    ?conflict drone:hasLongitude ?longitude.
    ?conflict drone:hasAltitude ?altitude.
    ?conflict drone:hasType ?type.
    ?conflict rdf:type ?ADBSConflict.
    FILTER(?altitude<500)
    FILTER(?type="UAV"^^xsd:string)
}
```

Listing 1. SPARQL statement to retrieve conflicts received by ADS-B and flying in VLL

Therefore, Dronetology-cas integration avoids unnecessary processing delays, as shown in the simulation results obtained, and also it simplifies maneuvers when one or more conflicts are dismissed.

2) *Scenario 2: Using NOTAM knowledge:* A CAS usually considers conflicts detected by collaborative or/and non-collaborative onboard transceivers or sensing devices. However other risks are known such as those listed in NOTAMs. NOTAMs are usually considered when planning an UAS flight but they may change so any subsequent update may not be considered. In the simulated scenario proposed, Dronetology-cas translates external data from a NOTAM provider to standard conflicts triplets that can be used by DAIDALUS during the UAS flight. The download of NOTAMs requires connectivity to external providers. As NOTAMs refresh rate is unknown we configure a periodical update of the NOTAM data stored in the KB.

The simulation performed consist of an UAS crossing a no-flight zone that it is defined in a NOTAM published during the flight so it has not been taken into account to plan the UAS flight, as depicted in Figure 11. DAIDALUS configuration for this simulation avoids altitude changes in any of the corrective

guidance generated, therefore, only horizontal maneuvers are considered. A *conflict-generator* software is developed to translate any NOTAM into multiple conflicts in the boundary of the no-flight zone. Conflicts generated are inserted in the KB. To avoid unnecessary processing, the *conflict-generator* only translates NOTAMs that are near the waypoints of the flight plan of the UAS.

The number of conflicts derived from a NOTAM and their positions are estimated using the default configuration of DAIDALUS (the CD3D parameters). The main parameter used is the Horizontal Miss Distance threshold (HMD) to separate generated conflicts so the *conflict-generator* avoids creating an excessive number of conflicts, so the no-flight zone boundary is partially covered.

During the simulated flight, the *conflict-generator* inserts conflicts considering the new NOTAM published and the KB provides DAIDALUS conflicts in the boundary of the no-flight zone, so DAIDALUS calculates a maneuver that changes the path of the UAS, avoiding the no-flight zone, as shown in the red dotted line in Figure 11. Therefore, Dronetology-cas integration increases situational awareness of the UAS as it allows to translate hazards, like no-flight zones, to conflicts that can be managed by DAIDALUS.

Listing 2 depicts the SPARQL query used to retrieve conflicts, including the ones generated from the NOTAM, in order to modify the UAS path and bypass a no-flight zone.

```
PREFIX rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns#>
PREFIX owl: <http://www.w3.org/2002/07/owl#>
PREFIX rdfs: <http://www.w3.org/2000/01/rdf-schema#>
PREFIX xsd: <http://www.w3.org/2001/XMLSchema#>
PREFIX drone: <http://www.dronetology.net/dronetology-cas.owl#>
```

```
SELECT ?latitude ?longitude ?altitude
WHERE
{
  ?conflict a drone:Conflict.
  ?conflict drone:hasLatitude ?latitude.
  ?conflict drone:hasLongitude ?longitude.
  ?conflict drone:hasAltitude ?altitude.
  FILTER(?altitude<500)
}
```

Listing 2. SPARQL statement to retrieve conflict data

IX. CONCLUSION

In this paper, we describe the Dronetology-cas ontology, as a value-added component for any CAS, and its integration with a specific one, DAIDALUS. Dronetology-cas integration modes allows multiple configurations for different purposes, such as being a black box or a conflict knowledge manager. Self developed software components integrate Dronetology-cas with DAIDALUS to provide knowledge. Results obtained from simulations performed show that knowledge available in the KB improves DAIDALUS capabilities and performance, reducing unnecessary processing and increasing safety. This was achieved inserting ADS-B data and NOTAM information as conflicts in the KB and defining SPARQL sentences to fetch conflicts. The installation of Dronetology-cas in a production-ready integration on a SBC should take into account the

performance results and the integration mode required to balance response time and memory consumption as obtained in the simulations performed.

Future work will be focused on improving performance, testing different SBC implementations, customizing the default linux-based OS, considering others, like Microsoft Windows IoT Core, and building a specific OS distribution devoted for CAS applications.

Another line of work is to create a *dataset* with semantic mission data to be used for research of UAS air traffic. Software components developed for the simulation could be evolved to a generic software simulation for UASs. Further developments of this work will develop an extension of Dronetology-cas to manage other UAS subsystems, like the path planner. These developments have the potential to expand the initial Dronetology-cas domain and to achieve an ontology standard for autonomous UAS.

REFERENCES

- [1] D. Martín-Lammerding, J. J. Astrain, and A. Cordoba, "A reference ontology for collision avoidance systems and accountability," in *SEMAMPRO 2021, The Fifteenth International Conference on Advances in Semantic Processing*, IARIA, 2021, pp. 22–28.
- [2] C. Muñoz, A. Narkawicz, G. Hagen, J. Upchurch, A. Dutle, M. Consiglio, and J. Chamberlain, "Daidalus: Detect and avoid alerting logic for unmanned systems," in *2015 IEEE/AIAA 34th Digital Avionics Systems Conference (DASC)*, IEEE, 2015, 5A1–1.
- [3] ICAO. (2021). Annex 13 — aircraft accident and incident investigation, [Online]. Available: <https://www.icao.int/safety/airnavigation/aig/pages/documents.aspx> (visited on 05/30/2022).
- [4] C. Rekkas and M. Rees, "Towards ads-b implementation in europe," in *2008 Tyrrhenian International Workshop on Digital Communications-Enhanced Surveillance of Aircraft and Vehicles*, IEEB, 2008, pp. 1–4.
- [5] B. Albaker and N. Rahim, "A survey of collision avoidance approaches for unmanned aerial vehicles," in *2009 international conference for technical postgraduates (TECHPOS)*, IEEE, 2009, pp. 1–7.
- [6] J. Phillips, J. Martinez, I. A. Bârsan, S. Casas, A. Sadat, and R. Urtaşun, "Deep multi-task learning for joint localization, perception, and prediction," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, IEEE, 2021, pp. 4679–4689.
- [7] W. W. Group. (2021). SSN, Semantic Sensor Network Ontology, <https://www.w3.org/tr/vocab-ssn/>, [Online]. Available: <https://www.w3.org/TR/vocab-ssn/> (visited on 05/30/2022).
- [8] R. M. Keller, S. Ranjan, M. Y. Wei, and M. M. Eshow, "Semantic representation and scale-up of integrated air traffic management data," in *Proceedings of the International Workshop on Semantic Big Data*, 2016, pp. 1–6.
- [9] H. Rahman and M. I. Hussain, "A light-weight dynamic ontology for internet of things using machine learning technique," *ICT Express*, 2020.
- [10] M. P. Owen, A. Panken, R. Moss, L. Alvarez, and C. Leeper, "Acas-xu: Integrated collision avoidance and detect and avoid capability for uas," in *2019 IEEE/AIAA 38th Digital Avionics Systems Conference (DASC)*, IEEE, 2019, pp. 1–10.
- [11] J. T. Davies and M. G. Wu, "Comparative analysis of acas-xu and daidalus detect-and-avoid systems," *National Aeronautics and Space Administration NASA Ames Research Center; Moffett Field CA United States Technical Report NASA/TM-2018-219773 ARC-E-DAA-TN50499*, 2018.
- [12] B. Zhou, P. Krähenbühl, and V. Koltun, "Does computer vision matter for action?" *Science Robotics*, 2019.
- [13] Redcat Holdings. (2021). Drone Box, <https://www.redcatholdings.com/drone-box>, [Online]. Available: <https://www.redcatholdings.com/drone-box> (visited on 05/30/2022).
- [14] TL-Elektronik. (2021). Black box, <https://www.tl-elektronik.com/>, [Online]. Available: https://www.tl-elektronik.com/index.php?page=uav&p_id=40&lang=en (visited on 05/30/2022).

- [15] UAV Navigation. (2021). Black Box <https://www.uavnavigation.com/>, [Online]. Available: <https://www.uavnavigation.com/sites/default/files/docs/2021-03/UAV%20Navigation%20FDR01%20Brochure.pdf> (visited on 05/30/2022).
- [16] D. Zuehlke, N. Prabhakar, M. Clark, T. Henderson, and R. J. Prazenica, "Vision-based object detection and proportional navigation for uas collision avoidance," in *AIAA Scitech 2019 Forum*, 2019, p. 0960.
- [17] U. Papa, G. Ariante, and G. Del Core, "Uas aided landing and obstacle detection through lidar-sonar data," in *2018 5th IEEE International Workshop on Metrology for AeroSpace (MetroAeroSpace)*, IEEE, 2018, pp. 478–483.
- [18] U. Papa, "Sonar sensor model for safe landing and obstacle detection," in *Embedded Platforms for UAS Landing Path and Obstacle Detection*, Springer, 2018, pp. 13–28.
- [19] N. Gellerman, M. Mullins, K. Foerster, and N. Kaabouch, "Integration of a radar sensor into a sense-and-avoid payload for small uas," in *2018 IEEE Aerospace Conference*, IEEE, 2018, pp. 1–9.
- [20] A. Muraru, "A critical analysis of sense and avoid technologies for modern uavs," *Advances in Mechanical Engineering ISSN: 2160-0619*, vol. 2, Mar. 2012. DOI: 10.5729/ame.vol2.issue1.23.
- [21] D. Martín-Lammerding. (2021). Dronetology, the UAS Ontology, <https://dronetology.net/dronetology>, [Online]. Available: <https://dronetology.net/dronetology> (visited on 05/30/2022).
- [22] —, (2021). Dronetology-cas, the anti-collision ontology, <https://dronetology.net/dronetology-cas>, [Online]. Available: <https://dronetology.net/dronetology-cas> (visited on 05/30/2022).
- [23] Web Working Group. (2021). SPARQL Query Language for RDF, <https://www.w3.org/2001/sw/wiki/sparql>, [Online]. Available: <https://www.w3.org/TR/sparql11-query/> (visited on 05/30/2022).
- [24] —, (2021). Spin Working Group, Rules for SPARQL, <https://www.w3.org/submission/spin-sparql/>, [Online]. Available: <https://www.w3.org/2001/sw/wiki/spin-sparql/> (visited on 05/30/2022).
- [25] T. Licu, F. Cioran, B. Hayward, and A. Lowe, "Eurocontrol—systemic occurrence analysis methodology (soam)—a "reason"-based organisational methodology for analysing incidents and accidents," *Reliability Engineering & System Safety*, vol. 92, no. 9, pp. 1162–1169, 2007.
- [26] Web Working Group. (2021). Web Ontology Language (OWL), <https://www.w3.org/owl/>, [Online]. Available: <https://www.w3.org/owl/> (visited on 05/30/2022).
- [27] —, (2021). Resource Description Framework (RDF), <https://www.w3.org/2001/sw/wiki/rdf>, [Online]. Available: <https://www.w3.org/rdf/> (visited on 05/30/2022).
- [28] Raspberry Pi Foundation. (2021). Raspberry Pi 3, <https://www.raspberrypi.org/>, [Online]. Available: <https://www.raspberrypi.org/> (visited on 05/30/2022).
- [29] D. Martín-Lammerding. (2021). Competency questions in sparql, <https://dronetology.net/sim/competency-questions.zip>, [Online]. Available: <https://dronetology.net/sim/competency-questions.zip> (visited on 08/02/2021).
- [30] S. Graham, J. De Luca, W.-z. Chen, J. Kay, M. Deschenes, N. Weingarten, V. Raska, and X. Lee, "Multiple intruder autonomous avoidance flight test," in *Infotech@ Aerospace 2011*, 2011, p. 1420.
- [31] UAVionix. (2021). Inert and alert: Intelligent ads-b for uas nas integration, [Online]. Available: <https://uavionix.com/downloads/whitepapers/Inert-and-Alert-CONOPS.pdf> (visited on 05/30/2022).
- [32] I. Frolow and J. H. Sinnott, "National airspace system demand and capacity modeling," *Proceedings of the IEEE*, vol. 77, no. 11, pp. 1618–1624, 1989.

Automatic Recognition of Continuous Sign Language for Public Services

Robson Silva de Souza

*Dept. of Computer Engineering and Industrial Automation
School of Electrical and Computer Engineering
Campinas, SP, Brazil
robsonnddesouza@gmail.com*

José Mario De Martino

*Dept. of Computer Engineering and Industrial Automation
School of Electrical and Computer Engineering
Campinas, SP, Brazil
martino@unicamp.br*

Janice Gonçalves Temoteo Marques

*Dept. of Human Development and Rehabilitation
Faculty of Medical Sciences, University of Campinas
Campinas, SP, Brazil
janicetm@unicamp.br*

Ivani Rodrigues Silva

*Dept. of Human Development and Rehabilitation
Faculty of Medical Sciences, University of Campinas
Campinas, SP, Brazil
ivanirs@unicamp.br*

Abstract—In this article, we present an automatic image recognition approach for assisting the communication between deaf people and hearing physicians. The aim of the approach is to help the interaction and exchange of information during medical interviews and in different public services, such as police departments, hospitals, and citizen service centers. Its scope is the automatic recognition of the continuous signing through the analysis of traditional video and depth data (RGB-D data). Recognition is performed by a cascade of two neural networks. First, a convolutional neural network encodes the visual input and extracts relevant features. Second, a recurrent neural network learns the mapping of the extracted features and transforms them into words. We use the Connectionist Temporal Classification approach to train the recurrent network with videos of different lengths and word sequences. Experiments on two continuous sign language datasets show the effectiveness of our approach, achieving an accuracy of around 91% in the Brazilian Sign Language (Libras) dataset and 94% in Greek Sign Language (GSL) in signer-independent continuous sign language setup.

Index Terms—Brazilian Sign Language; Sign language recognition; Continuous signing; long short term memory; connectionist temporal classification

I. INTRODUCTION

This paper builds upon and extends our previous work [1], where we present our approach, experiments, and results considering a Brazilian Sign Language (Libras) dataset in the context of anamnesis. The current paper extends the original one and also presents experiments and results, taking into account a Greek Sign Language (GSL) dataset of the public service interaction domain.

Anamnesis and clinical examination are the standard procedures of physicians to diagnose diseases and health problems of their patients. Anamnesis is a process of interviewing the patient to collect information about his/her current health complaints and medical history. The precise disclosure, correct understanding, and assessment of this information are preconditions for an effective diagnosis and the identification of the appropriate therapy. However, the effectiveness of the medical

interview is jeopardized if the physician and the patient do not have a common language for communication. That is usually the case when we consider a deaf patient who has sign language as his/her first language and does not master the written language of the physician who, by his/her side, does not understand sign language. A common solution to overcome this problem is to have a sign language interpreter assisting the deaf patient during the interview. Besides the operational difficulties of organizing an interpreter, another important drawback is the uncomfortable situation created by the introduction of a third party in the medical interview. During a medical interview, the patient should feel comfortable enough to share very personal and sensitive information, providing any and all relevant information to help the doctor make a correct diagnosis. A solution to overcome this potential breach of patient-doctor confidentiality is to provide a robust computer-based solution to support the communication between physicians and deaf patients. Although the interaction between doctor and patient is a two-way process, in this article, we focus only on the issue of automatic recognition of continuous signing based on computer-based recognition of video imagery. Our main focus of interest is the continuous signing recognition of the Brazilian Sign Language. However, to provide evidence that our approach can be successfully applied to other sign languages, we also evaluated our approach using a Greek Sign Language dataset available publicly [2].

Sign languages convey information by the movement of the hands, body, and face. They are perceived by vision. There is not a single, universal sign language used worldwide by deaf people. Each country has its own sign language [3]. The sign language of a country is independent of its oral language. For example, Deaf Americans speak the American Sign Language (ASL), the Deaf in the UK use the British Sign Language (BSL), and Deaf Australians speak the Australian Sign Language (Auslan). Deaf Brazilians use the Brazilian Sign Language (Libras).

There has been increasing research interest in automatic sign language recognition in recent years. Automatic sign language recognition applies computer vision combined with machine learning techniques to analyze and translate, into a written form, videos with sign language content.

The development of robust automatic sign language recognition systems is challenging. Several techniques have been proposed for automatic sign language recognition for a variety of sign languages, including the Brazilian Sign Language (Libras). Most efforts, however, have been limited to the study of isolated sign recognition, postures representative of cardinal numbers (0 to 10), and the manual alphabet or fingerspelling. Research on continuous signing recognition is still rare.

Concerning the representation of the input data, early works in automatic recognition of sign language commonly were based on hand-crafted features that are designed beforehand by human experts to extract a given set of chosen characteristics [4]. These features were used with architectures such as Hidden Models of Markov (HMM) [5] or Conditional Random Field (CRF) [6] for sequential modeling.

In the last years, deep learning techniques have made significant advances in computational vision due to the huge increase in computational power using graphical process units (GPUs), which have added to the availability of datasets with millions of images [7], [8], [9]. From this perspective, research on continuous sign language recognition also benefited, especially concerning the alignment of frame sequences to word sequences, be it acting on systems using depth networks solely [10] or combined with HMMs [11].

The challenge faced is a somewhat overlooked problem in which the sequences of glosses generally are available but not their time limits in the videos. In order to solve this, recent models based on Recurring Neural Network (RNN) with Connectionist Temporal Classification (CTC) [12] have reached the state of the art in this task [10], [13].

Even with all the advances, work in the area still has limitations, such as the recognition system processing only cropped sequences of the hand. However, a robust system must also take non-manual expressions into account, which are fundamental components of all sign languages [14]. Another limitation is to provide extra information about the signs to the recognition system, such as medium lengths of the signs on the videos or the developments of subsystems for particular parts of the body.

This article presents a method for automatic continuous sign language recognition of Libras during medical interviews. In addition, we also verified the generalization of our method on a Greek Sign Language (GSL) dataset [2]. Applying the method, we implement an approach based on Deep Learning that is capable of finding and using extracted data from signing from full-frame sequences. Therefore, it aligns sequences of video frames displaying continuous sign language content to sequence glosses. A gloss is a word (or a couple of words) of a written language that is consistently used to label a sign within the corpus, regardless of the meaning of that sign in a particular context or whether it has been systematically

modified in some way [15]. As pointed out in [16], glosses are a convenient way to write down the meaning of a sign, as they use written language to represent the signs.

The main contributions of this article are:

- The construction of a robust and representative dataset, composed of RGB information and depth of signage in Libras in order to contribute to the advancement of the research in this area.
- Execution of a Depth-Wise Separable Convolutional Network (DWSCN) based architecture, as feature extractor preprocessor. Insofar as we know, we are the first to employ this type of architecture in continuous sign language recognition systems.
- The development of a new architecture of sequential learning, based on Recurrent Neural Networks and Connectionist Temporal Classification, which learn to find and store relevant data in its memory cells from the full-frame sequences, without importing in its subsystems structures that process image patches.

The remainder of the paper is organized as follows: Section II contains a review of relevant related work. Section III presents our approach. Section IV describes the experiments performed, and Section V presents the conclusions.

II. RELATED WORK

The recognition of continuous signing is a far more complex task than the recognition of isolated signs, requiring more sophisticated methods to deal with the dynamics of production and the transition between signs. As previously stated, continuous signing recognition systems are more appropriate for real-world scenarios of interpersonal communication. However, it is observed that there is still little research that seeks to solve this problem. In the following paragraphs, we present approaches aimed at recognizing continuous signing based on computer vision.

Until recently, research was based on hand-crafted features for spatio-temporal representations in combination with sequence modeling methods. Among these, we can mention the algorithms based on threshold models such as [17]. Yang and colleagues focus on parameters of threshold models for labels of epenthetic movements, in which they perform tests using HMM, calculating the similarity between the sign model and the test sequence [17]. Techniques based on dynamic time distortion (DTW) were also widely used, which measure the similarity between two sequences of temporal data based on the minimum distance between them; therefore, the data have their lengths altered in order to obtain the best mapping. Among these works are those of Zhang and colleagues [18], that use DTW for the recognition of Chinese Sign Language (CSL) sentences using the Kinect. In [19], an HMM is used to learn hand and elbow features. The authors propose thresholds that describe the probability of removing transitional motion from a given video segment and use DTW to determine the endpoint for each candidate sign. The final recognition is obtained by concatenating the most possible signs.

Research using deep learning models has increased considerably in recent years, whether acting in independent systems or combined with HMMs in the then called hybrid approach. Using a German Sign Language recognition dataset called RWTH-PHOENIX-Weather, Koller and colleagues [20], [21], [22], [23] built a hybrid architecture, consisting of a convolutional neural network (CNN) to learn representations of frame-by-frame labels of hand-cut sequences and HMMs to model time dependencies. They trained a network with frame sequences from a label assignment initialization called flat-start. In [24] they use state alignment per frame, provided by an HMM as frame labeling to train the neural networks.

Although HMMs have achieved good results in several tasks that involve sign language, [25] indicates that traditional approaches to Markov models are limited because their states must be designed from a modest-sized state space, and that the dynamic program algorithm used to perform efficient inference with HMMs scales in time with quadratic time complexity $O(S^2)$ [26]. Moreover, Graves and colleagues claim that these models require assumptions that observations in HMM are independent to make inference treatable [12].

On the other hand, RNNs show great capacity for sequential learning. According to [12], they are not like hybrid systems, which inherit the previously mentioned inconveniences of HMMs. Furthermore, hybrid RNN-HMM systems are not able to exploit the full potential of RNNs for sequence modeling.

Another promising model is the CTC Network, originally proposed for speech recognition [12]. CTC is an ideal method for tasks where data is poorly labeled, i.e., it does not require *a priori* alignments between input and output sequences and allows recurring networks to be trained with different video lengths and label sequences.

Camgoz and colleagues propose an approach that breaks down the problem of recognizing signs into a series of expert systems called subunits [27]. Each subunit consists of three layers of neural networks: a Convolutional Neural Network (CNN) for extraction of spatial features; a Bidirectional Long Short-Term Memory (BLSTM) [28], an extension of LSTM [29] that temporarily models the features; and a loss layer based on the CTC. A recent work, [11], also uses CNN and LSTM but encapsulated it in an HMM model following the hybrid approach used in his previous work, this time exploring sequential parallelism to learn sign language, mouth shapes, and hand shape classifiers.

The works [10], [30]–[33] use CNNs as feature extractors, a 3D CNN model, or a 3D residual convolutional network (3D-ResNet). For modeling and sequential learning, they use Dilated Convolutional Networks or RNNs such as LSTM, Gated Recurrent Unit (GRU) [34] and their variants in combination with the CTC algorithm. Among these approaches, [13] is the one that achieved the best performance in the RWTH-PHOENIX-Weather dataset and also in a set of images captured by the Kinect called CSL-25K, which covers 100 daily life sentences expressed in Chinese Sign Language (CSL).

In our proposal, we also use recurrent neural networks with

CTC, but differently from the other approaches, we apply a depth-wise separable convolutional network that contains far fewer parameters and is computationally cheaper than the state-of-the-art convolutional neural networks, as for example, VGG16 [35], ResNet50 [36], and InceptionV3 [37].

III. METHOD

In this section, we present the guidelines for the construction of a dataset and our approach for recognizing continuous signing. The approach includes a CNN-based model for features extraction and an RNN architecture for learning the spatial-temporal dependencies that exist between the sentence signs. To solve the alignment problem between the probability sequences in the RNN outputs with the sequences of glosses, we used CTC.

A. Dataset construction

A dataset composed of Libras sentences related to medical interviews is fundamental for developing and testing our approach. No publicly available image databases of continuous signing in Libras have been found.

Through the study of existing datasets of other sign languages [38], [39] and with the intent of meeting our objectives, we developed specifications to be followed for the construction of our dataset. The proposal is to develop a robust dataset that simulates the internal environment of a clinic with artificial lighting, in which the deaf volunteer or interpreter performs the sign naturally.

Fig. 1 shows the execution flow. Thereafter, each module will be described in detail.

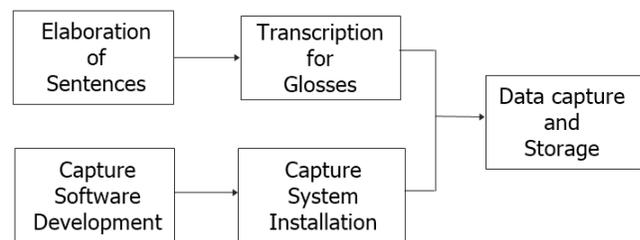


Fig. 1. Execution flow for the construction of the dataset

Sentences elaboration. Comprised of the elaboration of sentences in the Portuguese language from the answers of a patient in the context of a medical consultation (general practitioner). The sentences are established through the study of signs and manifested individual symptoms accordingly to the anamnesis medical procedure described in [40] and [41]. Following this procedure, our protocol encompasses:

- Main complaint: Brief phrase from the patient that explains his reason for looking for the physician;
- History of current sickness: Description of the main complaint concerning the chronology, when symptoms first manifested;

Transcription of the sentences in Portuguese to glosses. Glosses created through the assistance of a fluent sign lan-

guage specialist. The right columns on Tables I and II present the transcriptions of the sentences from the previous stage.

TABLE I
EXAMPLES OF SENTENCES DEVISED IN PORTUGUESE LANGUAGE AND ITS TRANSCRIPTIONS TO GLOSSES.

#	Target	Prediction
1	Eu tenho febre	EU FEBRE
2	Eu estou fraco	EU FRACO
3	Eu estou com diarreia	EU TER DIARRÉIA
4	Eu tenho manchas no rosto	EU TER MANCHA-ROSTO
5	Eu estou com tosse	EU TOSSE
6	Meu braço direito dói	MEU BRAÇO-DIREITO DOR
7	Meu braço esquerdo dói	MEU BRAÇO-ESQUERDO DOR
8	Meu dente dói	MEU DENTE DOR
9	Meu olho direito está vermelho	OLHO-DIREITO APONTAR VERMELHO TER
10	Meu olho esquerdo está vermelho	OLHO-ESQUERDO APONTAR VERMELHO TER
11	Minha urina está marrom	MEU XIXI COR MARROM
12	Minha boca está sangrando	MINHA BOCA SANGUE TER
13	Começou a um dia	COMEÇAR UM-DIA PASSADO
14	Começou há uma hora	COMEÇAR UMA-HORA PASSADO
15	Começou agora	COMEÇAR AGORA
16	Começou anteontem	COMEÇAR ANTEONTEM
17	Começou no domingo passado	COMEÇAR DOMINGO PASSADO
18	Faz um ano	COMEÇAR JÁ TER TEMPO UM ANO
19	Faz um mês	COMEÇAR JÁ TER TEMPO UM MÊS
20	Começou na quarta-feira passada	COMEÇAR QUARTA-FEIRA PASSADO

Capture device and development of the capture software.

The data recording is made through the Kinect device v2 for Windows. The capture application is developed using Kinect's own software development kit (SDK). This application captures and stores RGB images, depth images, and mapped images (RGB images mapped on the depth images), in which all pixels not belonging to the signer are converted to black.

Installation of the capture system. The Libras signing recordings executed by the volunteer are made in a laboratory, with artificial illuminations and homogeneous scene background.

The Kinect is fixed on an adjustable photographic tripod and positioned at approximately 1,2m high and to a distance of around 1,3m from the signer, as seen in figure 2. These positions are determined taking in consideration the capture hardware characteristics as, i.e., minimum distance (0,5m) and

TABLE II
EXAMPLES OF SENTENCES DEVISED IN PORTUGUESE LANGUAGE AND ITS TRANSCRIPTIONS TO GLOSSES - VERSION IN ENGLISH

#	Target	Prediction
1	I have fever	ME FEVER
2	I am weak	ME WEAK
3	I have diarrhea	ME HAVE DIARRHEA
4	I have spots on the face	ME HAVE SPOT-FACE
5	I have spots on the face	ME COUGH
6	My right arm hurts	MY RIGHT-ARM PAIN
7	My left arm hurts	MY LEFT-ARM PAIN
8	My tooth hurts	MY TOOTH PAIN
9	My right eye is red	RIGHT-EYE POINT RED HAVE
10	My left eye is red	LEFT-EYE POINT RED HAVE
11	My urine is brown	MY PEE BROWN COLOR
12	My mouth is bleeding	MY MOUTH BLOOD HAVE
13	It started a day ago	START ONE-DAY PAST
14	It started an hour ago	START ONE-HOUR PAST
15	It started now	START NOW
16	It started the day before yesterday	START THE-DAY-BEFORE-YESTERDAY
17	It started last Sunday	START SUNDAY LAST
18	It is been a year	START ALREADY HAVE TIME ONE-YEAR
19	It is been a month	START ALREADY HAVE TIME ONE-MONTH
20	It started last Wednesday	START WEDNESDAY LAST

maximum depth (4,5m), horizontal (70 degrees), and vertical (60 degrees) field of view.

The Kinect is connected to a computer with USB 3.0 port, with 64-bit (x64) operational system, physical dual-core 3.1 GHz processor, 4GB random access memory (RAM) or more, and a graphics adapter with DirectX 11 support [42].

In addition, the environment also has a conventional monitor (2D), in which the sentences to be signed are exhibited to the volunteer during the recordings.

Capture and data storage. The only request to the volunteer is to wear plain shirts of any color other than black, as some black shirt dyes can absorb infrared light, impairing Kinect's capacity of tracking the user (depth data) [42]. A Libras interpreter teacher member of the research team helps with the video acquisition. The position where the volunteer must be positioned during the recordings is indicated by ground marks. This position, called rest position, consists of standing in front of the capture device with lowered hands, as shown in Figure 2.



Fig. 2. Positioning and relative distances of the signer to the device.

Once the volunteer is properly positioned, the sentences are exhibited on the monitor to the volunteer. After each exhibited sentence, the volunteer must sign it naturally and return to the rest position after signing it. During the signing, the images are captured and stored on the computer. The participant is asked to remain in the rest position for a few seconds so the capture system can finish the data storage. The next sentence is then exhibited on the monitor, and the recording procedure as described above is repeated.

Figure 3 illustrates the process of capture and data storage.

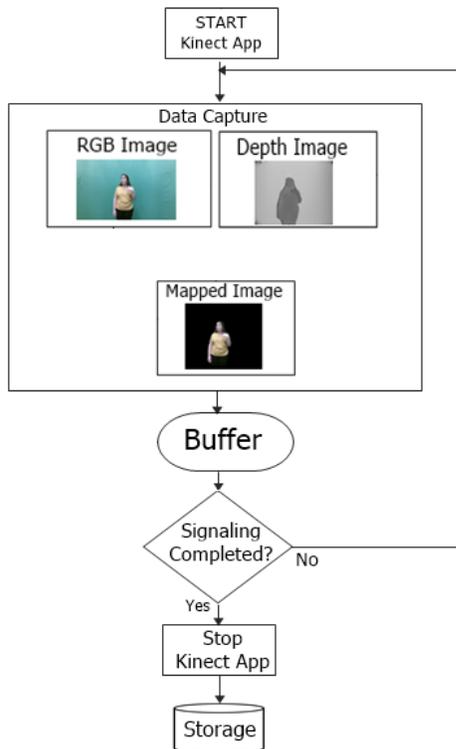


Fig. 3. Process of capture and data storage.

B. Our approach

The approach that recognizes continuous sign language includes a CNN-based model for features extraction and an

RNN architecture for learning the spatial-temporal dependencies that exist between the sentence signs. To solve the alignment problem between the probability sequences in the RNN outputs with the sequences of glosses, we used CTC.

Fig. 4 presents a general view of our approach composed of three main models. The first comprises spatial modeling, while the others encompass sequential learning and a CTC loss layer to decode categorical probabilities in sequences of glosses.

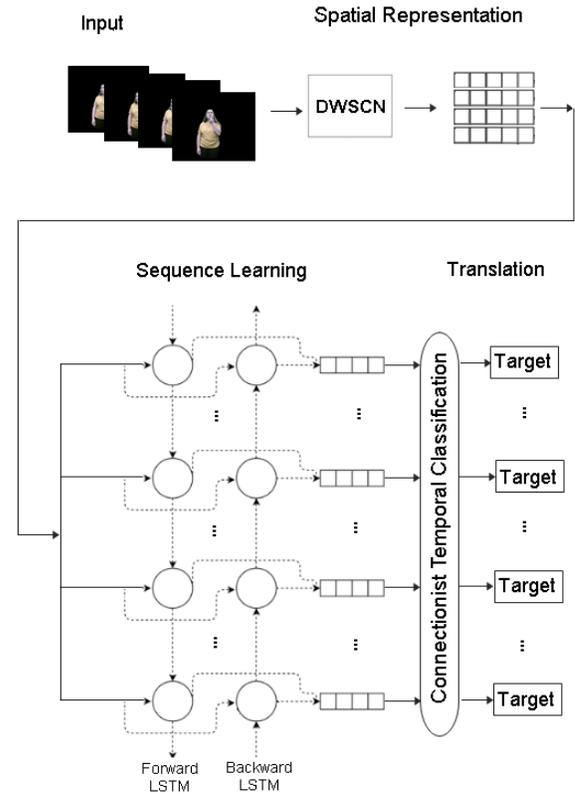


Fig. 4. Overview of our continuous sign language recognition approach

Features extraction. DWSCN is used for representations of spatial features of the frame sequences. The pre-trained MobileNetV1 [43] operational model is among the models based on the DWSCN. The use of pre-trained models enables the development of efficient models in situations of limited data availability, in addition to reducing processing time [44].

MobileNetV1 was pre-trained on ImageNet [7]. MobileNetV1 has a reduced size (17MB) and reduced number of parameters (4,2 million) when compared to other state-of-the-art models. Figure 5 shows the components of this architecture, in which each convolutional layer is followed by Batch Normalization and ReLU activation function.

MobileNetV1 expects color input images with size of $224 \times 224 \times 3$. Thus, the obtained images with Kinect must be rescaled to this size.

The pixel values are scaled between 0 and 1, and then each channel is normalized with respect to the ImageNet dataset according to (1), (2) and (3), where p_r , p_g and p_b are the

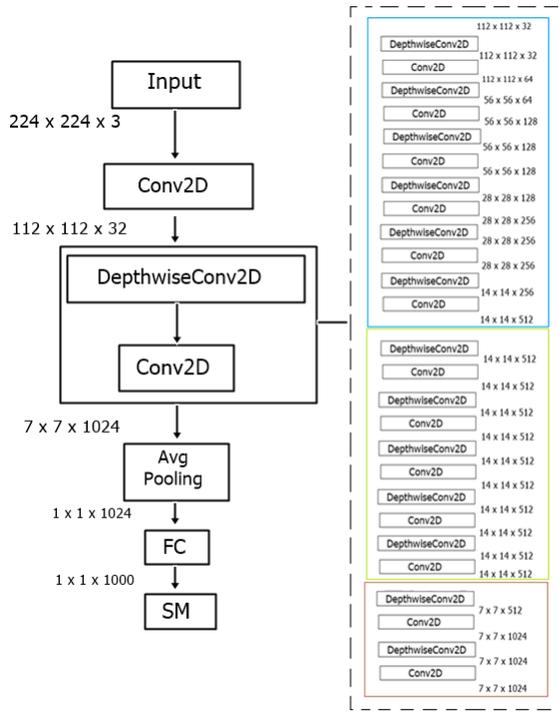


Fig. 5. MobileNetV1 Architecture

pixel values for the red, green and blue channels of the dataset images.

$$band_r = \frac{p_r/255 - mean_r}{std_r} \quad (1)$$

$$band_g = \frac{p_g/255 - mean_g}{std_g} \quad (2)$$

$$band_b = \frac{p_b/255 - mean_b}{std_b} \quad (3)$$

The averages $mean_r$, $mean_g$ and $mean_b$ with respect to the Imagenet are 0.485, 0.456, 0.406, respectively. The standard deviations std_r , std_g and std_b are equal to 0.229, 0.224, 0.225.

To use MobileNet as a feature extractor preprocessor, the softmax classification layer (SM) and the completely connected layer (FC) have been removed, keeping all the depth-wise separable convolution blocks and the Average Pooling layer.

All dataset images are processed by the resulting model. As the last layer has 1024 nodes, each image will be represented as a 1024 value vector. Each video sample results in a three-dimensional array of dimensions equal to 1 x number of frames x 1024 features. Since the numbers of frames are different between the videos, the padding in each array has been performed to allow the concatenating of all feature arrays.

All arrays are then concatenated and stored for a single NumPy array [45] in the standard binary file format (NPY).

The glosses are coded in categorical variables and, together with the feature arrays, are used as input to train our model

based on recurring neural networks. This is an overlooked learning problem as the gloss sequences are available but not its time limits.

Sequential learning. Our approach uses BLSTM to model the correspondences between the input sequences and output glosses. This architecture is capable of storing data for long periods of time and try to avoid the explosion of the gradient, a common problem of the Vanilla Neural Networks.

To implement a BLSTM network, it takes two parallel layers of LSTM cells, backward LSTM and forward LSTM, each of them being responsible for processing the information in the direction of time. The final hidden layer is given by the concatenation of the two networks.

The memory neurons of an LSTM are called cells. Fig. 6 presents the structure of a BLSTM network and highlights one single memory cell. The cells are capable of storing data in the course of a sequence through units called gates. According to [46], these units calculate the weights that connect them to avoid the gradient degradation through parameterized or manually chosen values.

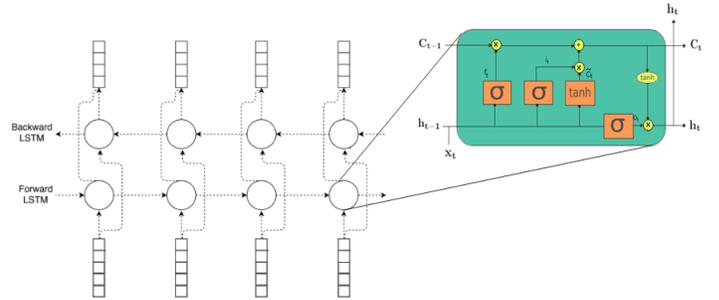


Fig. 6. BLSTM network structure, highlight to a single memory cell

The memory cell gates are composed of a sigmoid activation function and a multiplication operation between the weights and the inputs given by the Hadamard product. The operations that happen inside of an LSTM cell is detailed thereafter: The forget gate given by (4) decides which elements from the memory cell of previous state, C_{t-1} , are discarded.

$$f_t = \sigma(W_f.[h_{t-1}, x_t] + b_f) \quad (4)$$

The input gate given by (5) selects which information is going to be stored, multiplying its result by the candidate of the current memory cell, given by (6).

$$i_t = \sigma(W_i.[h_{t-1}, x_t] + b_i) \quad (5)$$

$$\tilde{C}_t = \tanh(W_c.[h_{t-1}, x_t] + b_c) \quad (6)$$

The hidden state of the current memory cell is given by (7), which combines the previous operations, that is, the process of forgetting $f_t * C_{t-1}$ and the process of insertion of new information on memory cell, $i_t * \tilde{C}_t$.

$$C_t = f_t * C_{t-1} + i_t * \tilde{C}_t \quad (7)$$

After these procedures, the hidden state of the current cell is built, h_t , given by (9), multiplying the output gate, (8), by the value of the hidden state of the memory cell, C_t .

$$o_t = \sigma(W_o \cdot [h_{t-1}, x_t] + b_o) \quad (8)$$

$$h_t = o_t * \tanh(C_t) \quad (9)$$

In the equations above, W_f , W_i , W_c , W_o , b_f , b_i , b_c , b_o are the values of the network weights and bias.

A softmax activation function on a fully connected layer is used in the network output and is applied to each time frame.

Connectionist Temporal Classification. In the BLSTM training phase, CTC is used to calculate the cost value. During prediction, it decodes the probability matrices of the softmax function in gloss sequences.

To allow the CTC algorithm to decode the target sequence, one more unit is introduced to the total number of labels in the softmax output layer. This unit refers to a token named blank, that models the transitions between different labels.

Let us consider the mapping of the input frames sequence $X = [x_1, x_2, \dots, x_T]$, for the sequences of output words $Y = [y_1, y_2, \dots, y_T]$. The CTC cost function for a pair (X, Y) has the conditional probability $p(Y/X)$ equal to the sum of all the valid paths $A \in A_{XY}$, calculating the probability $p_t(a_t|X)$ to a single step-by-step alignment following (10).

$$p(Y/X) = \sum_{A \in A_{XY}} \prod_a^b p_t(a_t|X) \quad (10)$$

For a training set M, the model parameters are tuned to minimize the negative log-likelihood. That way, the CTC objective function is given by (11).

$$Loss_{CTC} = \sum_{(X,Y) \in M} -\log p(Y/X) \quad (11)$$

To calculate the CTC loss efficiently, the Forward-Backward algorithm given in [12] is used.

IV. EXPERIMENTS

This section reports on the experiments performed and the performance of our architecture in continuous Libras signing recognition. We evaluated the performance of our method on the GSL dataset [2].

A. Datasets

In order to develop and test our approach, 280 sentences signed in Libras by a professional interpreter were captured, corresponding to 5 repetitions of 56 sentences. 42663 frames were obtained at a rate of 30 fps. The statistical details are presented in Table III.

TABLE III
STATISTICS OF OUR DATASET

Statistics	Data
Sentences	280
Vocabulary	67
Frames	42663
Glosses per Sentence	2 - 6
Frames per Sentence	124 - 277

RGB images obtained by Kinect have a resolution of 1920 x 1080 pixels. But for better performance of our data capture and storage application, at run time, these images were rescaled to 640 x 360 pixels. Table IV summarizes the written data and its corresponding sizes.

TABLE IV
SPECIFICATIONS RESULTING FROM OUR DATA CAPTURE AND STORAGE APPLICATION.

Data	Size
RGB Image	640 x 360
Depth Image	512 x 424
Mapped Image	512 x 424

Figure 7 presents 3 types of captured data: 7(a) Depth images, 7(b) RGB images and 7(c) Mapped images. In this paper, only mapped images are used.

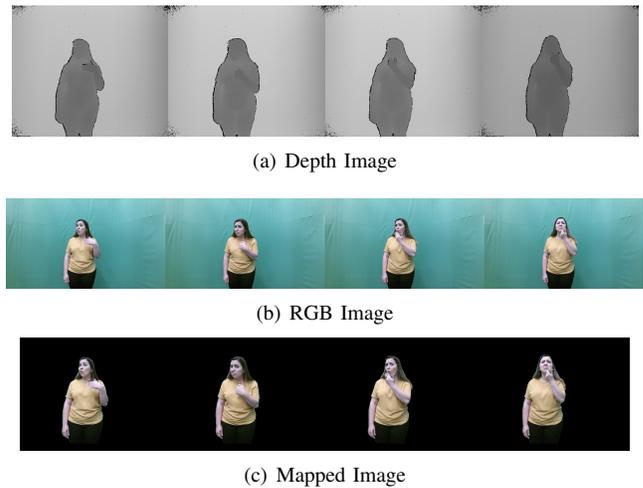


Fig. 7. Examples of data captured by the Kinect device.

We also evaluated our method on the GSL, in the Signer-independent continuous sign language recognition setup [2].

The GSL dataset is publicly available, and the captured data involve cases of deaf people interacting with different public services (police departments, hospitals, and citizen service centers). Recordings are performed in the laboratory using an

Intel RealSense D435 RGB+D at a rate of 30 fps. Data is acquired at a spatial resolution of 848x480 pixels. In total, 7 signers participate in data capture. Fig. 8 illustrates examples of frames from the GSL dataset.



Fig. 8. Example frames of the GSL dataset

The statistical details are presented in Table V:

TABLE V
STATISTICS OF GSL DATASET

Statistics	Data
Sentences	10295
Vocabulary	310
Frames	1036155
Mean sentence length	4,23
Frames/Sentence	6 - 615

B. Evaluation Metrics

The Word Error Rate (WER) is the metric widely [30], [31], [47], [48], [33], [32], [10], [13], [27], [21], [22], [23] used in continuous sign language recognition work. We also use this metric in our evaluation. The WER is given by (12).

$$WER = \frac{I + D + S}{N} \quad (12)$$

Where I is the number of errors entered, D is the number of deletion errors, S is the number of substitution errors, and N is the total number of glosses in the reference sentence.

The accuracy is given by (13).

$$acc = 1 - WER \quad (13)$$

C. Training and Evaluation

We performed experiments on an Nvidia RTX 2080Ti, and the model is implemented in the Keras framework [49], using tensorflow [50] as a backend.

Both in the Libras dataset and in the GSL dataset the simulations performed processed the mapped images. Initially,

these images were resized to 224 X 224 pixels, dimensions expected by the MobileNetV1 network, using bilinear interpolation. Then the images are converted to Numpy array, the pixel values are scaled between 0 and 1, and then each channel is normalized in relation to the ImageNet dataset.

After spatial modeling with our structure based on DWSCN, the resulting feature matrix has a dimension equal to the number of samples x time steps x features.

Libras dataset. In our experiments, we used 80 percent of the data (224 sentences) for the training set and 20 percent for the test set (56 sentences).

According to [51], training small datasets offers some challenges, as the network effectively memorizes the training dataset. The author recommends that adding noise is an approach to improve the generalization error and to enhance the structure of the mapping problem during learning. Thus, we applied Gaussian noise, at the entrance of the BLSTM network, with a standard deviation of 0.5 during the training phase.

The training of our BLSTM architecture is performed by implementing the backpropagation algorithm through time, [52]. The initialization for the recurrent weights matrix is orthogonal [53], for non-recurring weights the glorot uniform [54] and the vector bias is initialized with zeros. The optimizer used is the Root Mean Square Propagation (RMSprop) [55] with a learning rate of 0.01, a discounting factor of 0.9, a momentum of zero, (default values in the framework), and a batch size of 82. Then, we use the CTC beam decoder described in [56] to decode sentences with a beam width of 10.

For the aforementioned configurations, dozens of experiments were carried out using different network topologies, with a maximum of 4 layers (1 to 3 recurring layers and a completely connected layer) and the number of neurons equal to powers of 2 in the range of 2 to 512. The last layer is fixed with 68 neurons (one for each vocabulary label plus the blank label). Given the stochastic nature of the algorithms used, repetitions of the tests are performed in order to determine the most promising models.

In order to detect overfitting and determine the most promising models, a validation set is prepared, based on the training set, consisting of 60 sentences. During the training, at the end of each epoch, the value of the loss CTC is calculated in the validation set, and the best model in each training is determined according to the lowest value of the loss in that set.

Also, to identify and soften the effect of overfitting, we used the method of regularization called dropout, presented in [57]. Dropout values equal to 0.5 were applied for both recurrent and non-recurrent connections.

Among the best models that fit the data, the simplest model, that is, with the least hyperparameters, is considered the most plausible to be used in the test set.

GSL dataset. To verify the generalization, we also evaluate our method in the GSL dataset in the Signer-independent continuous sign language recognition setup [2]. The recordings

of a signer are separated into the validation set (588 sentences) and test set (881 sentences). The recordings of the other 6 signers comprise the training set (8821 sentences).

Due to memory constraints, we do not use all video frames of the GSL. Instead, we calculate the average of existing frames in the training set and use this value as the maximum amount for each video. As a result, all videos with more than 100 frames are reduced to 100, in such a way that the selected frames are determined from a linear spacing. Videos with less than 100 frames remain unchanged.

The random weight distribution types for the recurrent weights matrix, non-recurring weights, and the vector bias are the same as used in the Libras dataset.

Simulations were performed using the RMSProp optimizer and also with the optimizer Adaptive with Momentum (Adam) [58] and its variants Adadelta [59], Adagrad [60] and Adamax [58].

The hyperparameters of the optimizers' exponential decay rate for the 1st moment and the 2nd moment estimates were set at 0.9 and 0.999 respectively, while the learning rate varied from 0.01 to 0.00001. The batch size values in the simulations were: 128, 256, 512 and 1024. We use the CTC beam decoder to decode sentences with beam width of 100.

In the same way as we did in the Libras dataset, we used several topologies with a maximum of 5 layers (2 to 4 recurrent layers and a fully connected layer) and the number of neurons equal to a power of 2 in the range of 32 to 1024.

In order to reduce overfitting, in addition to applying the regularization dropout, we use the L2 regularization factor [61], varying its value from 0.1 to 0.0001.

D. Results

Libras dataset. Our best result was achieved by configuring two recurrent layers with 32 and 64 neurons, respectively. At the end of 30000 epochs, it was determined that the best model corresponds to epoch 21422. The values of the initial weights and the settings referring to that model were saved and stored for reproducibility, as well as for use in the unseen data set during the training.

Of the 56 sentences in the test set, 11 obtained some kind of error in the model prediction. The average WER was 8.92% and therefore, an accuracy of 91.07%. In Table VI we can observe some errors found, comparing the results of the model with the ground-truth sentences. Bold words are associated with errors in prediction. Table VII presents the equivalent results in English.

Therefore, the errors found were: 13 substitutions, 2 insertions, and no deletions. Low values in relation to the total amount of glosses existing in the dataset demonstrate the effectiveness of our architecture.

GSL dataset. Our best result is achieved by configuring two recurrent layers with 256 and 256 neurons, respectively. At the end of 40,000 epochs, it was determined that the best model corresponds to epoch 28371. This is achieved using the Adam optimizer, with a learning rate of 0.0001 and L2

TABLE VI
SENTENCES WITH PREDICTION ERRORS

#	Target	Prediction
1	COMEÇAR ANTEONTEM	COMEÇAR ONTEM
2	COMEÇAR QUINTA-FEIRA PASSADA	COMEÇAR TERÇA-FEIRA PASSADA
3	COMEÇAR SEGUNDA-FEIRA PASSADA	COMEÇAR QUARTA-FEIRA PASSADA
4	COMEÇAR TERÇA-FEIRA PASSADA	COMEÇAR QUINTA-FEIRA PASSADA
5	MAU-HÁLITO FEDOR TER	MAU-HÁLITO FEDOR VERMELHO TER
6	MEU DENTE DOR	MEU COSTAS TER
7	MEU NARIZ DOR	MEU OLHO-ESQUERDO INCHADO
8	OLHO-DIREITO APONTAR VERMELHO TER	OLHO-DIREITO APONTAR VERMELHO SABOR NÃO-TER
9	MEU OLHO-DIREITO DOR	MEU OLHO-DIREITO INCHADO
10	MEU OMBRO-DIREITO DOR	MEU PESCOÇO DOR
11	MEU PESCOÇO DOR	MEU GARGANTA MARROM

TABLE VII
SENTENCES WITH PREDICTION ERRORS - VERSION IN ENGLISH

#	Target	Prediction
1	START BEFORE-YESTERDAY	START YESTERDAY
2	START THURSDAY PAST	START TUESDAY PAST
3	START MONDAY PAST	START WEDNESDAY PAST
4	START TUESDAY PAST	START THURSDAY PAST
5	BAD-BREATH BAD-SMELL HAVE	BAD-BREATH BAD-SMELL RED HAVE
6	MY TOOTH PAIN	MY BACK HAVE
7	MY NOSE PAIN	MY LEFT-EYE SWOLLEN
8	RIGHT-EYE POINT RED HAVE	RIGHT-EYE POINT RED FLAVOR DO-NOT-HAVE
9	MY RIGHT-EYE PAIN	MY RIGHT-EYE SWOLLEN
10	MY RIGHT-SHOULDER PAIN	MY NECK PAIN
11	MY NECK PAIN	MY THROAT BROWN

regularization factor in each recurring layer with a value of 0.001.

Of the 881 sentences in the test set, 790 sentences were correctly predicted, and 91 sentences were obtained some kind of error in the model prediction. These 91 sentences total 372 glosses, of which 227 gloss predictions are correct, and the errors found were: 79 substitutions, 16 insertions, and 50 deletions. Therefore, the average WER was 6.0% and, consequently, the accuracy 94.0%.

In table VIII, we quantitatively compare our results with the best results obtained in [2]. In this work the authors implement recent deep neural network methods for continuous sign language recognition. Such methods are: SubUNets [27], GoogLeNet+TConvs [10], 3D-ResNet+BLSTM [31] and I3D+BLSTM [62]. Sequence alignment and decoding use CTC [12], Entropy Regularization CTC [63] and Stimulated CTC [64].

TABLE VIII
COMPARISON WITH METHODS ON GSL IN THE SIGNER-INDEPENDENT
CONTINUOUS SIGN LANGUAGE RECOGNITION SETUP

Method	WER (%)
SubUNets+CTC	20.58
3D-ResNet+BLSTM+EnStimCTC	24.01
GoogLeNet+TConvs+EnCTC	6.75
I3D+BLSTM+EnStimCTC	6.1
DWSCN+BLSTM+CTC (Our)	6.0

Compared to the mentioned methods, our method achieves the best performance in the test set in the GSL dataset in the Signer-independent continuous sign language recognition setup. This is achieved using only the traditional CTC criterion (computationally cheaper) rather than using extensions to it. Furthermore, there is no need to pretrain the model in the respective isolated sign dataset version in our method. Another advantage is that we do not need data augmentation techniques, and we use less data than what we provide.

V. CONCLUSIONS

In this article, we presented an approach for the recognition of continuous sign language. This approach receives sequences of images of a person communicating in sign language and translates continuous signing into written language. Our approach produces state-of-the-art comparable results.

In general, when compared to other approaches in the literature, our approach demonstrates a series of advantages:

- i) It does not depend on the extraction of manual features, specifically designed for a domain and laboriously calculated from the geometry of the hands and arms.
- ii) It takes into account characteristics related to non-manual expressions, such as movements of the face, eyes, head, and torso, instead of using only continuous sequences of the hands.

- iii) Contrary to other studies' continuous signing recognition, which performs the feature extraction process in video segments related to isolated signs, our spatial representation module is processed on the entire video. Our choice is due to the fact that video representation based on fixed-length signs can compromise the continuous recognition of signing in real situations since the same sign varies in length in a video, even when performed by the same person in different situations

- iv) Our spatial modeling, which is based on depthwise separable convolutions, reduces the latency and favors the development of real-time sign recognition because of the accuracy and the number of parameters and demanded calculations. This is a great advantage when compared to other convolutional neural networks.

- v) Our architecture based in BLSTM with CTC learns to find and store information relevant memory cells from the data channels included in full-frame sequences. This is done without injecting subsystems in its structure that process image patches. Consequently, our approach presents a greater capacity for temporal learning compared to studies that import extra data in its system to ease the learning.

Our approach demonstrates the potential to be applied in signing recognition on heterogeneous backgrounds due to the use of Kinect, which performs the segmentation of the individual while capturing the depth and color of images. In our upcoming work, we intend to include more signage and diversify the recording scenarios of our dataset images, as well as increase the vocabulary in order to maximize the robustness of our recognition approach.

ACKNOWLEDGMENT

This study was financed in part by the Coordination for the Improvement of Higher Education Personnel - CAPES, Brazil - Finance Code 001.

REFERENCES

- [1] R. S. de Souza, J. M. De Martino, J. G. T. Marques, and I. R. Silva, "Automatic recognition of continuous signing of brazilian sign language for medical interview," in *Sixth International Conference on Informatics and Assistive Technologies for Health-Care, Medical Support and Wellbeing HEALTHINFO*, Barcelona, Spain, 2021.
- [2] N. M. Adaloglou, T. Chatzis, I. Papastratis, A. Stergioulas, G. T. Papadopoulos, V. Zacharopoulou, G. Xydopoulos, K. Antzakis, D. Papazachariou, and P. none Daras, "A comprehensive study on deep learning-based methods for sign language recognition," *IEEE Transactions on Multimedia*, 2021.
- [3] N. Timmermans *et al.*, *The status of sign languages in Europe*. Council of Europe, 2005.
- [4] L. Nanni, S. Ghidoni, and S. Brahmam, "Handcrafted vs. non-handcrafted features for computer vision classification," *Pattern Recognition*, vol. 71, pp. 158–172, 2017.
- [5] B. Bauer and H. Hienz, "Relevant features for video-based continuous sign language recognition," in *Proceedings Fourth IEEE International Conference on Automatic Face and Gesture Recognition (Cat. No. PR00580)*, pp. 440–445, IEEE, 2000.
- [6] J. D. Lafferty, A. McCallum, and F. C. Pereira, "Conditional random fields: Probabilistic models for segmenting and labeling sequence data," in *Proceedings of the Eighteenth International Conference on Machine Learning*, pp. 282–289, 2001.
- [7] J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, and L. Fei-Fei, "Imagenet: A large-scale hierarchical image database," in *2009 IEEE Conference on Computer Vision and Pattern Recognition*, pp. 248–255, Ieee, 2009.

- [8] O. Russakovsky, J. Deng, H. Su, J. Krause, S. Satheesh, S. Ma, Z. Huang, A. Karpathy, A. Khosla, M. Bernstein, *et al.*, “Imagenet large scale visual recognition challenge,” *International Journal of Computer Vision*, vol. 115, no. 3, pp. 211–252, 2015.
- [9] A. Krizhevsky, I. Sutskever, and G. E. Hinton, “Imagenet classification with deep convolutional neural networks,” in *Advances in Neural Information Processing Systems*, pp. 1097–1105, 2012.
- [10] R. Cui, H. Liu, and C. Zhang, “A deep neural framework for continuous sign language recognition by iterative training,” *IEEE Transactions on Multimedia*, 2019.
- [11] O. Koller, C. Camgoz, H. Ney, and R. Bowden, “Weakly supervised learning with multi-stream cnn-lstm-hmms to discover sequential parallelism in sign language videos,” *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2019.
- [12] A. Graves, S. Fernández, F. Gomez, and J. Schmidhuber, “Connectionist temporal classification: labelling unsegmented sequence data with recurrent neural networks,” in *Proceedings of the 23rd International Conference on Machine Learning*, pp. 369–376, ACM, 2006.
- [13] H. Zhou, W. Zhou, Y. Zhou, and H. Li, “Spatial-temporal multi-scale network for continuous sign language recognition,” *arXiv preprint arXiv:2002.03187*, 2020.
- [14] K. Brown, *Encyclopedia of language and linguistics*, vol. 1. Elsevier, 2005.
- [15] T. Johnston, “From archive to corpus: transcription and annotation in the creation of signed language corpora,” in *Proceedings of the 22nd Pacific Asian Conference on Language, Information, and Computation*, pp. 16–29, 2008.
- [16] A. Baker, B. van den Bogaerde, R. Pfau, and T. Schermer, *The linguistics of sign languages: An introduction*. John Benjamins Publishing Company, 2016.
- [17] W. Yang, J. Tao, and Z. Ye, “Continuous sign language recognition using level building based on fast hidden markov model,” *Pattern Recognition Letters*, vol. 78, pp. 28–35, 2016.
- [18] J. Zhang, W. Zhou, and H. Li, “A new system for chinese sign language recognition,” in *2015 IEEE China Summit and International Conference on Signal and Information Processing (ChinaSIP)*, pp. 534–538, IEEE, 2015.
- [19] J. Zhang, W. Zhou, and H. Li, “A threshold-based hmm-dtw approach for continuous sign language recognition,” in *Proceedings of International Conference on Internet Multimedia Computing and Service*, p. 237, ACM, 2014.
- [20] O. Koller, J. Forster, and H. Ney, “Continuous sign language recognition: Towards large vocabulary statistical recognition systems handling multiple signers,” *Computer Vision and Image Understanding*, vol. 141, pp. 108–125, 2015.
- [21] O. Koller, R. Bowden, and H. Ney, “Automatic alignment of hamnosys subunits for continuous sign language recognition,” *LREC 2016 Proceedings*, pp. 121–128, 2016.
- [22] O. Koller, H. Ney, and R. Bowden, “Deep hand: How to train a cnn on 1 million hand images when your data is continuous and weakly labelled,” in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 3793–3802, 2016.
- [23] O. Koller, O. Zargaran, H. Ney, and R. Bowden, “Deep sign: Hybrid cnn-hmm for continuous sign language recognition,” in *Proceedings of the British Machine Vision Conference 2016*, 2016.
- [24] O. Koller, S. Zargaran, and H. Ney, “Re-sign: Re-aligned end-to-end sequence modelling with deep recurrent cnn-hmms,” in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 4297–4305, 2017.
- [25] Z. C. Lipton, J. Berkowitz, and C. Elkan, “A critical review of recurrent neural networks for sequence learning,” *arXiv preprint arXiv:1506.00019*, 2015.
- [26] A. Viterbi, “Error bounds for convolutional codes and an asymptotically optimum decoding algorithm,” *IEEE Transactions on Information Theory*, vol. 13, no. 2, pp. 260–269, 1967.
- [27] N. C. Camgoz, S. Hadfield, O. Koller, and R. Bowden, “Subunets: End-to-end hand shape and continuous sign language recognition,” in *2017 IEEE International Conference on Computer Vision (ICCV)*, pp. 3075–3084, IEEE, 2017.
- [28] A. Graves and J. Schmidhuber, “Framewise phoneme classification with bidirectional lstm and other neural network architectures,” *Neural networks*, vol. 18, no. 5–6, pp. 602–610, 2005.
- [29] S. Hochreiter and J. Schmidhuber, “Long short-term memory,” *Neural computation*, vol. 9, no. 8, pp. 1735–1780, 1997.
- [30] J. Pu, W. Zhou, and H. Li, “Dilated convolutional network with iterative optimization for continuous sign language recognition,” in *IJCAI*, pp. 885–891, 2018.
- [31] J. Pu, W. Zhou, and H. Li, “Iterative alignment network for continuous sign language recognition,” in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 4165–4174, 2019.
- [32] R. Cui, H. Liu, and C. Zhang, “Recurrent convolutional neural networks for continuous sign language recognition by staged optimization,” in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 7361–7369, 2017.
- [33] H. Zhou, W. Zhou, and H. Li, “Dynamic pseudo label decoding for continuous sign language recognition,” in *2019 IEEE International Conference on Multimedia and Expo (ICME)*, pp. 1282–1287, IEEE, 2019.
- [34] K. Cho, B. Van Merriënboer, D. Bahdanau, and Y. Bengio, “On the properties of neural machine translation: Encoder-decoder approaches,” *arXiv preprint arXiv:1409.1259*, 2014.
- [35] K. Simonyan and A. Zisserman, “Very deep convolutional networks for large-scale image recognition,” *arXiv preprint arXiv:1409.1556*, 2014.
- [36] K. He, X. Zhang, S. Ren, and J. Sun, “Deep residual learning for image recognition,” in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 770–778, 2016.
- [37] C. Szegedy, V. Vanhoucke, S. Ioffe, J. Shlens, and Z. Wojna, “Rethinking the inception architecture for computer vision. 2015,” *arXiv preprint arXiv:1512.00567*, 2015.
- [38] N. B. Ibrahim, H. H. Zayed, and M. M. Selim, “Advances, challenges and opportunities in continuous sign language recognition,” *Journal of Engineering and Applied Sciences*, vol. 15, no. 5, pp. 1205–1227, 2020.
- [39] F. Quiroga, “Sign language recognition datasets.” https://facundoq.github.io/unlp/sign_language_datasets/index.html, 2017. Accessed 10 June 2020.
- [40] F. Veiga and A. B. Souza, *Physical Exam Manual*. Elsevier Brasil, 2019.
- [41] M. H. Swartz, *medical semiology treatise*. Elsevier Brasil, 2015.
- [42] M. Rahman, *Beginning Microsoft Kinect for Windows SDK 2.0: Motion and Depth Sensing for Natural User Interfaces*. Apress, 2017.
- [43] A. G. Howard, M. Zhu, B. Chen, D. Kalenichenko, W. Wang, T. Weyand, M. Andreetto, and H. Adam, “Mobilenets: Efficient convolutional neural networks for mobile vision applications,” *arXiv preprint arXiv:1704.04861*, 2017.
- [44] J. Brownlee, *Deep Learning for Computer Vision: Image Classification, Object Detection, and Face Recognition in Python*. Machine Learning Mastery, 2019.
- [45] T. E. Oliphant, *A guide to NumPy*, vol. 1. Trelgol Publishing USA, 2006.
- [46] I. Goodfellow, Y. Bengio, and A. Courville, *Deep Learning*. The MIT Press, 2016.
- [47] D. Guo, W. Zhou, H. Li, and M. Wang, “Hierarchical lstm for sign language translation,” in *Thirty-Second AAAI Conference on Artificial Intelligence*, 2018.
- [48] J. Huang, W. Zhou, Q. Zhang, H. Li, and W. Li, “Video-based sign language recognition without temporal segmentation,” in *Thirty-Second AAAI Conference on Artificial Intelligence*, 2018.
- [49] F. Chollet *et al.*, “Keras.” <https://keras.io>, last accessed on 02/08/21, 2015.
- [50] M. Abadi, A. Agarwal, P. Barham, E. Brevdo, Z. Chen, C. Citro, G. S. Corrado, A. Davis, J. Dean, M. Devin, *et al.*, “Tensorflow: Large-scale machine learning on heterogeneous distributed systems,” *arXiv preprint arXiv:1603.04467*, 2016.
- [51] J. Brownlee, *Better Deep Learning: Train Faster, Reduce Overfitting, and Make Better Predictions*. Machine Learning Mastery, 2018.
- [52] P. J. Werbos, “Backpropagation through time: what it does and how to do it,” *Proceedings of the IEEE*, vol. 78, no. 10, pp. 1550–1560, 1990.
- [53] A. M. Saxe, J. L. McClelland, and S. Ganguli, “Exact solutions to the nonlinear dynamics of learning in deep linear neural networks,” *arXiv preprint arXiv:1312.6120*, 2013.
- [54] X. Glorot and Y. Bengio, “Understanding the difficulty of training deep feedforward neural networks,” in *Proceedings of the Thirteenth International Conference on Artificial Intelligence and Statistics*, pp. 249–256, 2010.
- [55] G. Hinton, N. Srivastava, and K. Swersky, “Neural networks for machine learning lecture 6a overview of mini-batch gradient descent,” *Cited on*, vol. 14, p. 8, 2012.

- [56] A. Hannun, C. Case, J. Casper, B. Catanzaro, G. Diamos, E. Elsen, R. Prenger, S. Satheesh, S. Sengupta, A. Coates, *et al.*, "Deep speech: Scaling up end-to-end speech recognition," *arXiv preprint arXiv:1412.5567*, 2014.
- [57] G. E. Hinton, N. Srivastava, A. Krizhevsky, I. Sutskever, and R. R. Salakhutdinov, "Improving neural networks by preventing co-adaptation of feature detectors," *arXiv preprint arXiv:1207.0580*, 2012.
- [58] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," *arXiv preprint arXiv:1412.6980*, 2014.
- [59] M. D. Zeiler, "Adadelata: an adaptive learning rate method," *arXiv preprint arXiv:1212.5701*, 2012.
- [60] J. Duchi, E. Hazan, and Y. Singer, "Adaptive subgradient methods for online learning and stochastic optimization.," *Journal of machine learning research*, vol. 12, no. 7, 2011.
- [61] A. Krogh and J. A. Hertz, "A simple weight decay can improve generalization," in *Advances in Neural Information Processing Systems*, pp. 950–957, 1992.
- [62] J. Carreira and A. Zisserman, "Quo vadis, action recognition? a new model and the kinetics dataset," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 6299–6308, 2017.
- [63] H. Liu, S. Jin, and C. Zhang, "Connectionist temporal classification with maximum entropy regularization," *Advances in Neural Information Processing Systems*, vol. 31, pp. 831–841, 2018.
- [64] C. Wu, M. J. Gales, A. Ragni, P. Karanasou, and K. C. Sim, "Improving interpretability and regularization in deep learning," *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, vol. 26, no. 2, pp. 256–265, 2017.