IEEE
Computational Intelligence
MAGAZINE

AUGUST 2022
VOLUME 17 NUMBER 3
WWW.IEEE-CIS.ORG

IEEE Computational Intelligence Society

IEEE

IEEE DSAA 2022

The 9th IEEE International Conference on Data Science and Advanced Analytics

October 13-16, 2022 Shenzhen, China

The 9th IEEE International Conference on Data Science and Advanced Analytics (DSAA) features its strong interdisciplinary synergy among statistics (via ASA), computing and information/intelligence sciences (via IEEE and ACM), and cross-domain interactions between academia and business for data science and analytics. DSAA sets up a high standard for its organizing committee, keynote speeches, paper submissions to main conference and special sessions, and a competitive rate for paper acceptance. DSAA has been widely recognized as a dedicated flagship annual meeting in data science and analytics by Google Metrics, Australian CORE ranking, and China Computer Foundation.

DSAA'2022 provides a premier forum that brings together researchers, industry and government practitioners, as well as developers and users of big data solutions for the exchange of the latest theoretical advances in Data Science and the best practice for a wide range of applications.

The Highlights of DSAA' 2022

- Strong Research and Applications tracks with reproducible and open results.
- Student and Industry Poster sessions with lightning results highlighting research advances of students and best practices of industries.
- Journal tracks with Machine Learning and J. Data Science and Analytics
- Industrial Day with practitioners of leading global and local industries.
- Panel on the trends and controversies of data science and advanced analytics.
- Special Session on the foundations and emerging areas of data science.
- Strong interdisciplinary participation from analytics, machine learning, statistics, etc.
- Activities and events to foster cross-domain interactions among researchers, scientists, and practitioners from academia, industry, and government.
- Financially sponsored by IEEE CIS, proceedings by IEEE Xplore and EI indexed.
- Technically supported by ACM SIGKDD, ASA, and CCF.

Call for Papers

DSAA'2022 is a dual-track conference consisting of a Research Track and an Application Track. DSAA'2022 invites submissions of papers reporting innovative research on all aspects of data science and advanced analytics, as well as application-oriented papers that make original, significant, and reproducible contributions to improving the practice of data science and analytics. DSAA'2022 will also feature three Journal Tracks, a peer-reviewed Student Poster session, and an Industry Poster session.

Submission website: https://cmt3.research.microsoft.com/DSAA2022

Important Dates

Paper Submission	Special Session Submission	Notification of Acceptance	Special Sessions Notification
May 15, 2022	Jun 1, 2022	Jun 25, 2022	Jun 1, 2022

Sponsored by

IEEE · IEEE Computational Intelligence Society · ASA · acm · 中国计算机学会 · 深圳大学 SHENZHEN UNIVERSITY · 中国科学院深圳理工大学

General Chairs
Joshua Zhexue Huang
Gary Yen
Karolj Skala

Program Chairs – Research Track
Yi Pan
Barbara Hammer
Muhammad Khurram Khan

Program Chairs – Application Track
Xing Xie
Laizhong Cui

Special Session Chairs
Satoshi Kurihara
Qi Liu

Tutorial Chair
Xintao Wu

Industry Poster Chair
Yanchang Zhao

Student Poster Chair
Ju Fan

Trends & Controversies Panel Chair
Jianhua Huang

PhD Track Chair
Philippe Fournier-Viger

Best Paper Award Chairs
Ee-Peng Lim
Gillian Dobbie

Next-Generation Data Scientist Award Chair
Graham Williams

Local Arrangement Chair
Lei Zhang

Publicity & Media Chairs
Yulei Wu
Chuanren Liu
Bo Yang
Yipeng Zhou

Industry Track Chair
Yu Qiao
Wei Liu
Yanjie Wei

Registration Chair
Dingming Wu

Sponsor Chairs
Longbing Cao
Shu Yang

Proceeding Chair
Yulin He

Webmaster
Yaodong Huang

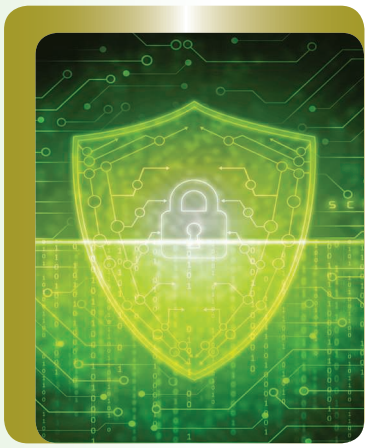
csse.szu.edu.cn/dsaa2022/

dsaa2022.dsaa.co

Digital Object Identifier 10.1109/MCI.2022.3182848

IEEE Computational Intelligence MAGAZINE

Volume 17 Number 3 ❑ August 2022
www.ieee-cis.org



on the cover
©SHUTTERSTOCK.COM/ADESIGNS

Features

Columns

Departments

IEEE Computational Intelligence Magazine (ISSN 1556-603X) is published quarterly by The Institute of Electrical and Electronics Engineers, Inc. **Headquarters:** 3 Park Avenue, 17th Floor, New York, NY 10016-5997, U.S.A. +1 212 419 7900. Responsibility for the contents rests upon the authors and not upon the IEEE, the Society, or its members. The magazine is a membership benefit of the IEEE Computational Intelligence Society, and subscriptions are included in Society fee. Replacement copies for members are available for US$20 (one copy only). Nonmembers can purchase individual copies for US$220.00. Nonmember subscription prices are available on request. Periodicals postage paid at New York, NY and at additional mailing offices. Postmaster: Send address changes to IEEE Computational Intelligence Magazine, IEEE, 445 Hoes Lane, Piscataway, NJ 08854-1331 U.S.A. PRINTED IN U.S.A. Canadian GST #125634188.

Digital Object Identifier 10.1109/MCI.2022.3182826

Digital Object Identifier 10.1109/MCI.2022.3180606

Chuan-Kang Ting
National Tsing Hua University, TAIWAN

Editor's Remarks

Quest for the Balance of AI and Privacy

The employment of machine learning often requires a large amount of data, including our sensitive and personal information, to train models for various applications. With growing worldwide concerns over data privacy, particularly with the recent introduction of the EU's General Data Protection Regulation (GDPR), it has become an important topic for AI research and application to implement data privacy-preserving technologies. Researchers or companies that utilize personal data in their AI models face the pressure and difficulty of balancing the advancement of AI and data privacy. In the quest for such a balance, machine learning technologies that conform to some particular aspects of data protection regulated by the GDPR, often combining privacy-preserving technologies such as homomorphic encryption, differential privacy, secure multiparty computation, or federated learning, are used to protect data and models from being accessed by other parties.

Two *Feature* articles in this issue give us examples of privacy-preserving technologies. The first article presents an introduction to deep learning based on homomorphic encryption for guaranteeing the privacy of user data. The second article proposes a decentralized federated learning framework assisted by blockchain to avoid information leakage from malicious clients. As demanding as complying with regulations may seem, privacy preserving will build trust between consumers and AI systems, further increasing the use and opportunity of AI.

The *Columns* comprises two articles about multi-objective optimization (MOO). The first article proposes the CSD indicator that assesses the convergence, sparsity, and diversity of solutions for sparse MOO. It compares the CSD values of 11 state-of-the art multi-objective evolutionary algorithms (MOEAs) on 60 test problems. The second article formulates pandemic containment as an MOO problem considering infection peaks, economic damage, and containment cost, and deals with the problem using MOEAs.

In the new Industrial and Governmental Activities column, the VP-IGA invites industry leaders to participate in workshops, conferences, and panels for the connection and collaboration between CIS and the industry. In this issue, we interview Yongduan Song, the new Editor-in-Chief of IEEE Transactions on Neural Networks and Learning Systems, who talks about this publication, his life and favorites. We hope you will enjoy the articles in this issue. Please do not hesitate to contact me at ckting@pme.nthu.edu.tw if you wish to give any feedback for our magazine.

Digital Object Identifier 10.1109/MCI.2022.3180649
Date of current version: 19 July 2022

President's Message

Jim Keller
University of Missouri, USA

AI by Any Other Name

Hi all,

To borrow from The Bard, what's in a name? The world is awash with the hype, the reality, the potential, and the concerns of Artificial Intelligence. But what constitutes AI? That depends on who, and when, you ask. When I was a kid (academically speaking), tasked with teaching a course on AI, I "knew" what it was: systems with propositions consisting of discrete symbols, manipulated through first order predicate calculus, sometimes with a helping of probability on the side. There were languages and hardware developed to assist in that enterprise. So, when the IEEE Neural Networks Council/Society changed its name to the IEEE Computational Intelligence Society, it made perfect sense to me that CI was different from AI since the underlying models were rooted in numeric and functional representations, producing a nice crisp partition of techniques.

But was that ever the case? Not long ago, I found a copy of the 1955 proposal by John McCarthy with Marvin Minsky, Nathaniel Rochester and Claude E. Shannon for the now famous Dartmouth conference in the Summer of 1956 and in which they coined the term "Artificial Intelligence" [1]. They defined AI as "the science and engineering of making intelligent machines." McCarthy et al conjectured "that every aspect of learning or any other feature of intelligence can in principle be so precisely described that a machine can be made to simulate it." But, there in the proposal, was the inclusion of the fledgling "neuron networks," a pillar of computational intelligence. So, the partition of methodologies and techniques into AI and CI was never crisp, and certainly with respect to neural networks was not even a fuzzy partition where membership is shared between the groupings—artificial neural networks is recognized as a full component of both AI and CI. Hence, (some of you at least knew this was coming), the partition in my mind is possibilistic, where any given approach can have varying degrees of typicality (membership) in both worlds. For example, is fuzzy logic restricted only to CI? Well, some of us argue that it represents one of the ways to address the growing field of XAI (eXplainable AI). Members of CIS went through an exercise a few years ago to try to determine the boundaries between AI and CI. We discovered that there was no consensus. Some held my early 1980s-fueled thought that AI and CI were disjoint, some that CI was a subset of AI (or vice versa), and many postulating varying degrees of overlap. I'm arguing here that we shouldn't get too hung up with names. We are the CIS, but we are also the society of AI, though many of our

Digital Object Identifier 10.1109/MCI.2022.3180650
Date of current version: 19 July 2022

Digital Object Identifier 10.1109/MCI.2022.3181716

sister societies share in this possibilistic partition. When, under the lead of CIS, a group of IEEE Societies proposed and established the *IEEE Transactions on AI*, we chose to write a simple scope "theories and methodologies of Artificial Intelligence" (along with applications). This allows us to accommodate the breadth and evolution of the field. Our founding EiC, Hussein Abbass, established some guiding principles for his term at the helm, but like the "pirate code," they are more guidelines than rules.

To wrap these ramblings up, I've at least decided to take a broad perspective about the membership of any given approach in the sets of CI and AI methods, believing that they are not mutually exclusive. Reminds me of a potential society tagline we discussed a while back: *the best AI is CI.* The main thing is to do good and responsible science and engineering, and to have fun. Feel free to contact me at kellerj@missouri.edu with your thoughts, suggestions, questions, and innovative ideas, even/especially if you disagree. Please stay safe and healthy. I hope to see many of you this year as we try to emerge from the pandemic.

Reference

[1] J. McCarthy, M. Minsky, N. Rochester, and C. Shannon, 1955, "A proposal for the Dartmouth summer research project on artificial intelligence," archived from the original on 2007-08-26, retrieved 10:47 (UTC), 9th of April 2006, reprinted in *AI Mag.*, vol. 27, no. 4, pp. 12–14, 2006.

Call for Papers for Journal Special Issues

Special Issue on "Machine Learning Assisted Evolutionary Multi-objective Optimization"
Journal: ***IEEE Computational Intelligence Magazine***
Guest Editors: Xingyi Zhang, Ran Cheng, Liang Feng, and Yaochu Jin
Submission Deadline: September 1, 2022
https://cis.ieee.org/images/files//Documents/call-for-papers/cim/CIM-SI-MLEMO_CFP.pdf

Special Issue on "Explainable Representation Learning-based Intelligent Inspection and Maintenance of Complex Systems"
Journal: ***IEEE Transactions on Neural Networks and Learning Systems***
Guest Editors: Zhigang Liu, Cesare Alippi, Hongtian Chen, and Derong Liu
Submission Deadline: September 1, 2022
https://cis.ieee.org/images/files/Documents/call-for-papers/tnnls/202202-Explainable_Representation_Learning-based_Intelligent_Inspection_and_Maintenance_of_Complex_Systems.pdf

Special Issue on "User Evaluation for VR Games"
Journal: ***IEEE Transactions on Games***
Guest Editors: Hai-Ning Liang, Wenge Xu, Yiyu Cai, and Fotis Liarokapis
Submission Deadline: September 1, 2022
https://transactions.games/submit/special-issues

Special Issue on "Cognitive Learning of Multi-Agent Systems"
Journal: ***IEEE Transactions on Cognitive and Developmental Systems***
Guest Editors: Yang Tang, Wei Lin, Chenguang Yang, Nicola Gatti, and Gary G. Yen
Submission Deadline: September 30, 2022
https://cis.ieee.org/images/files/Documents/call-for-special-issues/Cognitive_Learning_of_Multi-Agent_Systems-IEEE_TCDS-CFP-20211210.pdf

Special Issue on "Resource Sustainable Computational and Artificial Intelligence"
Journal: ***IEEE Transactions on Emerging Topics in Computational Intelligence***
Guest Editors: Joey Tianyi Zhou, Ivor Tsang, and Yew Soon Ong
Submission Deadline: February 1, 2023
https://cis.ieee.org/images/files/Publications/TETCI/SI26_CFP_RSCAI.pdf

Digital Object Identifier 10.1109/MCI.2022.3182851

Piero P. Bonissone
Piero P. Bonissone Analytics, USA

What is New in Industry?

Welcome to a new column of the IEEE Computational Intelligence Magazine devoted to Industrial Activities. In a previous communication[1], I described the vision for the newly formed IEEE CIS Industrial and Governmental Activities Committee (IGAC). With IGAC, we strive to provide services, products, and offerings of interest to Industry, ranging from webinars to tutorials, conferences, publications, standards, and other aspects of CIS activities, as illustrated in Figure 1, adapted from reference[1].

This issue's column will focus on *Industry Engagement in Conferences*. We want to offer industry-centered panels, workshops, and conferences with keynotes by Industry leaders. We wish to create a forum within CIS conferences, in which Industry can participate without necessarily having to submit scholarly papers. Industry leaders can define current and future challenges that AI/CI technology could address. We can achieve this goal with focused panels and keynote speeches. With increasing Industry participation in CIS conferences, we will grow conference exhibits, sponsorships, and, more importantly, guidance to make CI/AI technology relevant to Industry needs. In the sequel, we will describe two of our efforts in this direction.

Our first effort is the creation of a unique Industry-centered event within our flagship conference, the 2022 IEEE World Congress on Computational Intelligence (IEEE WCCI 2022). For the first time, we have established an *Industry Day* within the IEEE WCCI 2022 Program (https://wcci2022.org/industry-day/). This event, included in the IEEE WCCI 2022 regular registration, is also accessible, for a nominal fee, to non-WCCI participants.

During this inaugural *Industry Day*, which will take place in Padua, Italy, on July 20, 2022, we will offer a representative sample of industrial and commercial applications of CI and AI technologies. Industry Day's program will contain one keynote presentation, two special sessions, seven panels, and twelve short presentations from Industry. Furthermore, we expect to have over 100 application-oriented poster presentations throughout the day.

Industry Day's two special sessions will be devoted to *CI in Industry 4.0* and *CI in Metallurgy*. Specifically:

- *Industry 4.0* (i.e., the Fourth Industrial Revolution) covers cyber-physical systems, additive manufacturing, virtual and augmented reality, cloud computing, big data analytics, data science, etc. This session will describe the next level of manufacturing, where machines will redefine themselves in how they communicate and perform individual functions.
- *CI for Metals Science and Technology* covers the application and deployment of CI in metallurgy. This session will facilitate a dialogue among AI technology providers and end-users from the metal and alloy industry and explore further opportunities for cooperation.

Industry Day's seven panels will cover a variety of topics:

- *Outsmart Your Competitors*: Real Examples of Value Extraction From the Data Journey. Five representatives of the Swiss Data Innovation Alliance will share their best practices and real market experiences to demonstrate how to use data to extract

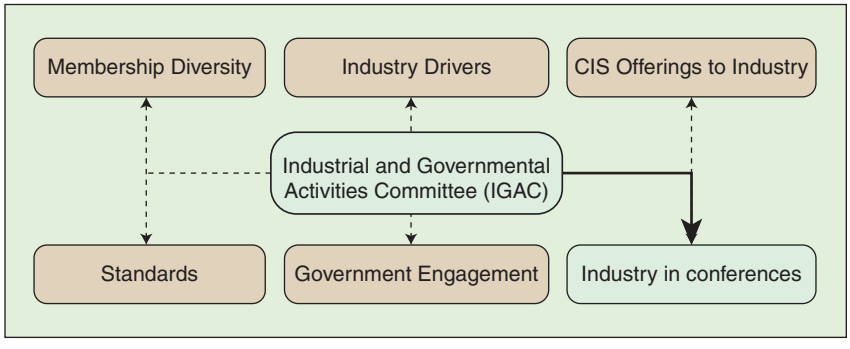


FIGURE 1 IEEE CIS Industrial and Governmental Activities Committee

[1]Piero P. Bonissone, "IEEE CIS VP Industrial and Governmental Activities Vision Statement", IEEE Computational Intelligence Magazine, 17(1):5-7, 2022.

Digital Object Identifier 10.1109/MCI.2022.3180846
Date of current version: 19 July 2022

… Create a forum within CIS conferences, in which Industry can participate without having to submit scholarly papers. Established Industry Day within the IEEE WCCI 2022 Program. Organizing the 2023 IEEE Conference on AI as an industry-centered conference.

value at each step of the data journey and be ahead of the competition.

- *AI-enabled Cybersecurity and Privacy*: This panel will share real-world challenges, novel applications, and the most recent advances in security AI, and privacy from theoretical and empirical perspectives. It will address questions such as: How can we make a system robust to novel or potentially adversarial inputs? How can machine-learning systems detect and adapt to changes in the environment over time? When can we trust that a system that has performed well in the past will continue to do so in the future?
- *Industrial AI*: This panel will cover industrial applications of AI, ranging from production to manufacturing, quality control, operations, and maintenance. We will show illustrative examples from large and small companies. We will extend the discussion beyond the underlying AI technology to cover Business and Technology Challenges. We will explore Business Models to monetize AI-improved processes or AI-enabled services or products; KPIs for AI-enabled process improvement; Productization of AI-enabled services and products; AI Technology development and deployment; Technology sustainability, such as Model maintenance, etc.
- *Technology and Regulatory Challenges for AI in Europe*: We will share the latest policy document on AI of the IEEE European Public Policy Committee ICT Working Group and discuss the technology and regulatory challenges for AI in Europe, hearing from experts from Academia, Industry, and EU Policymakers.
- *Applied Deep Learning in Satellite Images using Small Supervised Data*: Deep Learning technologies need to have annotated data to train their models. This issue is particularly severe in operational scenarios where it is not possible to annotate data in a reasonable time frame or in cases in which we do not have enough available data related to rare scenarios. This panel will deal with methodologies for training neural networks with little data, using Self-Supervised Learning techniques.
- *AI Techniques for Natural Language Processing (NLP)*: Questioning Evaluation and Resource consumption for Deep Language Models in real-world scenarios. Large Language Models (LMs), currently reaching a trillion parameters, need reliable, complete, and representative benchmarks. This task usually requires significant effort in text annotation, intensive in both time and required knowledge, except for simple cases. These benchmarks are far from real industrial NLP use cases. Moreover, some LMs provide limited benefits to NLP tasks, where sometimes cheap, known, ML techniques are as effective with a tiny fraction of cost/time/resources/carbon footprint. This panel will focus on how to understand in advance when such an effort can bring significant benefits.
- *AI for Earth Observation*: AI for Earth Observation (AI4EO) focuses on harnessing the power of AI with the vast amount of EO data now available. While today the new boost of AI4EO is mainly related to Computer Vision applied to high-resolution satellite imagery, there are many other areas for Earth Science, prediction and big data analytics that could benefit from AI for the deployment of practical applications. The panel will discuss some of the main challenges in AI for Earth Observation of interest for research and business communities, i.e., scalable big data analytics, trustworthy and explainable AI, physics-aware AI, self-learning AI, AI-based EO data fusion and prediction for Digital Twin Earth (DTE). Here are examples of issues that will be discussed: How to augment EO capability with AI? How to make the AI decision-making more transparent? How to integrate "first principles" and "domain knowledge" into the AI statistical approach? How to develop unsupervised learning for EO data without labels? How to leverage the combination of EO, models and AI techniques to support high-resolution prediction and informed decision making with DTE?

The interested reader can find additional information on this event at https://wcci2022.org/industry-day/.

Our second effort in industry-centered conferences is the organization of the 2023 IEEE Conference on AI (IEEE CAI) - https://cai.ieee.org/2023/. This conference, co-sponsored by the Computational Intelligence, Computer, Signal Processing, and Systems Man and Cybernetics Societies, will take place on June 7-8, 2023 in Santa Clara, CA, USA. IEEE CAI 2023 will be the first full-fledged conference following this new format. Different Industry segments have different needs, requirements, regulatory constraints, and levels of technology adoption. To reflect this heterogeneity, we will structure IEEE CAI along six verticals, covering *AI in Energy, Healthcare/Life Science, Transportation/Aviation, Earth System Decision Support, Industrial AI, and Social Implications of AI/Privacy*. As we get closer to that date, we will provide additional information in our next columns.

CIM Editorial Office

Career Profile

Interview With Editor-in-Chief of IEEE Transactions on Neural Networks and Learning Systems

The Editor-in-Chief of IEEE Transactions on Neural Networks and Learning Systems (TNNLS), Prof. Yongduan Song, talks about this reputable publication, his life and favorites.

1. What events in your life most likely placed you on the path that led you to where you are today?

There are too many such events (big or small) in my life to recall, but what I would like to share here is my viewpoint on success.

I believe different people would have different opinions about success. To me, any progress is success. There is but one secret to success—never give up and nothing is impossible to a willing heart. Dr. Marie Curie used to say, "We must have the perseverance and, above all, confidence in ourselves."

Enthusiasm and persistence for scientific research are the key to success for everyone during his/her career development. Failures are inevitable and full of the life, for instance, paper and proposal rejections are not uncommon in pursuit of professionalism. The key is to learn from mistakes and gain from failures. As we all know, "failure is the mother of success." In fact, when doing the same thing, some people succeed but some fail. Failure is not fearful; the important thing is how to face it correctly. Facing failure, people should never take their fate lying down but try their best to work harder and harder until at last they succeed. In addition, confidence is just the door to success. If someone wants to get into the house of success, determination is necessary. If someone is not determined, he would give up when facing difficulty. Young members should enjoy the process of constant trial and modification.

Yongduan Song on the meeting

The important thing in doing research is to have a clear goal and the determination to attain it. There is a famous motto saying that there are no two identical people in the world, but successful people are similar in many ways. Success should also rely on such factors as diligence, confidence, perseverance. They are the key to help us to open the door of success and the foundation and condition of success. There is no denying that diligence is the first step to the success. As the proverb says, "no pains, no gains". Many scholars become successful by their diligence, which should be taken as a good habit.

I am fortunate to have the opportunity to study and conduct research in several well-known universities and national laboratories in China, Australia and the United States, collaborating with renowned scholars and research teams, close interactions with many top research scientists and groups. What benefited me the most is my research stay at home and aboard, the excellence of the people around me brought me both pressure and motivation. I am grateful for the experience of working with many successful and excellent scholars who taught me the attitude and spirit of research.

2. Can you tell us a little bit about your research work that brought you to this appointment?

One of my research directions lies in exploring the functionality and reliability of neural networks (NNs) when connected in the loop of control systems. Several interesting results have been obtained along this topic. For instance, motivated by the practical biological traits, we proposed a novel strategy that allows the NN to evolve with time varying rather than constant ideal weights. To tackle the difficulty arising from the varying and unknown nature of the ideal weights, we proposed to update the upper norm of the unknown weights, instead of the weights themselves. To deal with the situation that on one hand the function to be approximated by NN must be continuous over the operation region and, on the other hand, the function might involve discontinuity due to

Digital Object Identifier 10.1109/MCI.2022.3180881
Date of current version: 19 July 2022

Future AI should be able to not only sift and extract valuable information (Learn) from complex and changing environments (Perceive) but also create new meanings (Abstract) and have the ability to assist humans in planning and deciding (Reasoning), while meeting human needs (Integration) and concerns (Ethics and Security).

unexpected disturbances or even subsystem failures suddenly occurring during system operation, we proposed to use the NN to reconstruct the upper norm bound of the function, rather than the function itself. Such treatment not only simplifies the overall computations, but also allows the discontinuity possibly involved in the function to be handled gracefully. With inspiration from nervous systems that involve abundant interconnected cells with a variety of complex mechanisms dealing with incoming signals in many different ways, we proposed to use diversified basis (activation) functions in constructing NN-based approximator. This is deemed useful and sensible because the activation function with fixed single structure, chosen from the commonly used sigmoidal function, hyperbolic tangent function, or raised cosine function, might not be a wise choice as each neural structure presents specific characteristics. Thus, for some complex systems undergoing a myriad of changes, monotonous structure is not able to meet the learning requirements. To address the tough question of "how large is sufficiently large" for the number of neurons in an NN unit in control design, we proposed a new structural NN unit with grouped neurons and self-adjusting number of subneurons, i.e., the unit starts with a finite number of neurons in each group and automatically adds or deletes certain number of neurons if needed, using a strategy developed by the tracking error related criteria, as motivated by the fact that the neurons in the neurological systems are updating themselves constantly.

My areas of research have been closely related to neural networks, control and the related learning systems. Being appointed as the Editor-in-Chief (EiC) for TNNLS is absolutely an honor and a huge responsibility, requiring dedication, commitment, enthusiasm and devotion. This position also demands strong leadership experience and management skill. I am energetic and enthusiastic in IEEE related services, and I consider myself articulate in communicating and interacting with the community, strongly self-motivated with professional dedication.

IEEE TNNLS is an internationally renowned journal, under the guidance and leadership of the President and VP for publication at CIS, I will work closely and diligently with the editorial team to maintain TNNLS at its top level and advance the journal to an even higher level of excellence.

3. Where will your research take you from here?

In collaborating with Jennie Si from Arizona State University, USA, and Sonya Coleman and Dermot Kerr from Ulster University, UK, we have recently put together a special issue for TNNLS on biologically inspired methodologies for sensing, control and decision making, which is actually the current focus of research in my team. We are particularly interested in innovations and technologies that allow engineered systems to achieve desired performance and be resilient to externally or internally caused errors and unpredictable failures. In essence, this requires engineered designs capable of learning and self-reconfiguring and awareness of themselves and the environment it operates. Therefore, reliable sensing and decision-making are needed in engineering systems. Traditional engineering approaches have taken into account these design considerations, but usually, the solutions are costly. Yet, biological organisms in nature have successfully demonstrated their superior capability of processing a large amount of information, dealing with uncertainties when perceiving and processing data of their surroundings, adapting to environmental changes, and recovering from their internal errors and failures. All of these essential attributes are desired by engineered systems. Therefore, it is expected that biologically learned and inspired methods may offer fundamentally new theoretical frameworks and new design approaches to address system robustness, reliability, and effectiveness. Some progress has been made in bio-inspired control methodologies and our next step of research is to further verify and demonstrate the benefits and efficiency of the developed algorithms and strategies.

4. AI has accomplished remarkable achievements and showed its benefits to human life; on the other hand, people are concerned about its potential risks. Can you share your thoughts about the development and future of AI?

Since the victory of AlphaGo over Lee Sedol and Ke Jie, it has been genuinely recognized that AI can surpass humans in specific areas. In fact, AI has made significant progress in some data concentrated fields (transportation, medical, finance, etc.); however, the high dependence of current AI algorithms on data literally limits its cross-field applicability. In the foreseeable future, AI will still mainly play the role of assisting humans but not replacing them, and the development of AI will be a long and tortuous process. I believe that the AI of the future should be able to not only sift and extract valuable information (Learn) from complex and changing environments (Perceive) but also create new meanings (Abstract) and have the ability to assist humans in planning and deciding (Reasoning), while meeting human needs (Integration) and concerns (Ethics and Security). I have expectations that

AI will develop into a "strong and friendly AI", i.e., entirely passing Turing test, conscious, capable of emotional expression, and at the same time minimizing its potential risks.

5. What is your vision for TNNLS in five years from now?

First of all I have several observations: 1) AI is everywhere, making NNs more and more popular and finding their wide spectrum of applications in almost everywhere; 2) NNs are undergoing a kind of renaissance, allowing NNs and machine learning to continue playing increasingly important impact on AI; 3) NNs and Learning-Driven AI will continue to be the main focus of research for decades to come; and 4) TNNLS has made remarkable history during the past decade, thanks to its great authorship, editorship, and leadership.

With its tremendous success, I believe that TNNLS is currently well positioned for further development. With AI being so popular and neural networks and Learning being so critical to AI development, TNNLS should actively seize this golden opportunity and position itself as the leading scientific journal in the world. In fact, TNNLS is at the forefront of AI; the journal will and should be a leading outlet for cutting-edge research, serving as a forum for disseminating important results to the advancement of neural and learning sciences.

A number of measures being taken by TNNLS to further promote its excellence include: inviting top notch scholars worldwide to help with organizing special issues and/or contributing influential papers; attracting more papers from different countries to choose TNNLS as their home journal; expanding the editorial board by including some Associate Editors (AEs) from the industry and government sector; attracting more high quality papers from mathematics, computer science, biology, neuroscience and application-oriented fields; exploring new dimensions of TNNLS by publishing newly emerging research areas to revitalize the field; expanding the outreach and targeting the wider scientific communities to promote TNNLS.

Yongduan Song enjoying the view in Australia

As the EiC, I will interact closely with the AEs to maintain a highly motivated and effectively performing editorial board, ensuring that TNNLS's review process is completed in a timely manner, with high prestige publishing research results and best quality, strengthening TNNLS's global authorship, and rendering its articles consistently ranked among the world's most cited research. Through its print and online publishing, we hope to significantly increase the number of worldwide readership for TNNLS.

Yongduan Song with the former TNNLS EiC Haibo He at an international conference

6. What advices would you like to offer to interested authors to submit their research work?

Two important aspects must be checked carefully during the preparation for a paper to be submitted to the journal: (1) whether it is in line with the scope of the journal and (2) whether the quality of the paper reaches the standard of the journal and the contribution of the work is significant enough for the journal consider its possible publication.

TNNLS is prestigious and very selective. A successful submission should have an interesting title and the contributions/innovations of the work

Everyone has potential to be successful, so young members should be self-confident and willing to make every effort to work toward success.

described therein should be clear, significant and convincing. The presentation should be easy to follow and make sure to have a thorough research review on the related works and the results are substantiated and highly related.

Five Minutes with Prof. Yongduan Song

7. What was your service pathway in the Computational Intelligence Society (CIS)?

My service pathway started from serving a volunteer helping with paper review, organizing CIS conference sessions, serving on the editorial board as Guest Editor and Associate Editor, to now as the EiC for TNNLS. It is really an honor and privilege to have such an opportunity to serve in CIS.

8. What is your typical working day?

Normally I start my day with a morning jog and a full gourmet breakfast; then I drive to the office starting my work by checking emails and prioritizing the planned "to do" lists. Ever since I became the EiC for TNNLS, I have developed the habit to take care of the editorial issues as the first thing to do every day. Perhaps like many other researchers, my working day does not end at the normal working hour, enriching our lives with haste and enjoyment.

9. What is your ideal weekend?

The weekend is a good time to spend with family, and if weather permits, I normally would go on a picnic with my family and enjoy some views. Sometimes I play tennis or badminton with my friends; going to the movies and eating at my favorite restaurant are also the choice for my ideal weekend.

10. Give one interesting fact about yourself.

One interesting fact about myself is that I am good at cooking, although I do not cook often. Whenever there is a party at home, I will always be the one doing the main dishes which are quite stylish and tasty.

In my spare time, I like to play Go. For me, every game is a play of life and there are a lot of life truths in Go. "One begets two, two begets three, and three begets everything". In Go, one is the board, two is the opponent, and three is the variation. It is the change that produces everything, which taught me an interesting way of viewing the world and the things surrounding it.

11. What are you reading, watching, or listening to at the moment?

With heavy scientific research and administrative tasks, I have relatively little free time to read classic novels. I normally read some short essays in Reader's Digest in my spare time. I listen to classical music through some playlists, which could gently put me in a calming state of mind. I also like some classic popular songs like jazz, rock, and country songs from the last century.

12. Can you share with us one success story that will motivate young members and provide useful guidelines for their careers?

I would like to share with young members my viewpoint on success. Everyone has potential to be successful, so young members should be first of all self-confident, and at the same time, be willing to make every effort to work toward success. They should dare to make mistakes, maybe the process of constant trial and modification is comparable to the development of intelligent algorithms. Dare to question, dare to innovate, and don't just dabble and stop. Highly successful people often experience a complex "intelligent learning" process and the optimal solution originates from courage, self-confidence and hardworking.

Profile: Yongduan Song

Professional qualifications:

- ❑ Ph.D. (EE), Tennessee Technological University, USA, 1992
- ❑ M.S. (EE), Chongqing University, China, 1985
- ❑ B.S. (EE), Sichuan University, China, 1983

Current position:

- ❑ Professor and Dean of the School of Automation, Chongqing University
- ❑ Founding Director of Research Institute for Artificial Intelligence, Chongqing University

Institutions or companies where you have taught/conducted research:

- ❑ North Carolina A&T State University; Eglin Air Force Base, FL, USA; Navy Research Base (Washington); NASA Langley Research Center; National Institute of Aeronautics (NIA, USA); Beijing Jiaotong University; Chongqing University; Interplanetary (Chongqing) Intelligent Equipment Technology Research Institute

Most notable award/recognition:

- ❑ IEEE Fellow
- ❑ Editor-in-Chief, IEEE Transactions on Neural Networks and Learning Systems

Yongduan Song, *Chongqing University, CHINA*
Jon Garibaldi, *University of Nottingham, UK*
Carlos A. Coello Coello, *CINVESTAV-IPN, MEXICO*
Georgios N. Yannakakis, *University of Malta, MALTA*
Huajin Tang, *Zhejiang University, CHINA*
Yew Soon Ong, *Nanyang Technological University, SINGAPORE*
Hussein Abbass, *University of New South Wales, AUSTRALIA*

CIS Publication Spotlight

IEEE Transactions on Neural Networks and Learning Systems

A Review of Single-Source Deep Unsupervised Visual Domain Adaptation, by S. Zhao, X. Yue, S. Zhang, B. Li, H. Zhao, B. Wu, R. Krishna, J. E. Gonzalez, A. L. Sangiovanni-Vincentelli, S. A. Seshia, and K. Keutzer, *IEEE Transactions on Neural Networks and Learning Systems*, Vol. 33, No. 2, Feb 2022, pp. 473–493.

Digital Object Identifier: 10.1109/TNNLS.2020.3028503

"Large-scale labeled training datasets have enabled deep neural networks to excel across a wide range of benchmark vision tasks. However, in many applications, it is prohibitively expensive and time-consuming to obtain large quantities of labeled data. To cope with limited labeled training data, many have attempted to directly apply models trained on a largescale labeled source domain to another sparsely labeled or unlabeled target domain. Unfortunately, direct transfer across domains often performs poorly due to the presence of domain shift or dataset bias. Domain adaptation (DA) is a machine learning paradigm that aims to learn a model from a source domain that can perform well on a different (but related) target domain. In this article, we review the latest single-source deep unsupervised DA methods focused on visual tasks and discuss new perspectives for future research. We begin with the

definitions of different DA strategies and the descriptions of existing benchmark datasets. We then summarize and compare different categories of single-source unsupervised DA methods, including discrepancy-based methods, adversarial discriminative methods, adversarial generative methods, and self-supervision based methods. Finally, we discuss future research directions with challenges and possible solutions."

IEEE Transactions on Fuzzy Systems

Variational Fuzzy Superpixel Segmentation, by T. C. Ng and S. K. Choy, *IEEE Transactions on Fuzzy Systems*, Vol. 30, No. 1, Jan 2022, pp. 14–26.

Digital Object Identifier: 10.1109/TFUZZ.2020.3029939

"This article presents a novel variational model based on fuzzy clustering and total variation regularization for superpixel segmentation. Compared with the classical hard-labeled methodologies, our approach gives soft results via the fuzzy membership function, and moreover, the use of total variation provides additional information that can enhance the superpixel regularity, which in turn improves the segmentation performance. To efficiently minimize the energy functional of the proposed model, we adopt an alternating direction method of multipliers with the modified Chambolle's fast duality projection algorithm. Our algorithm can generate regular and compact superpixels with high segmentation accuracy, satisfactory boundary adherence, and low computational cost. Comparative experimental results with the current state-of-the-art approaches reveal the superior performance of the proposed method."

How to Vary the Input Space of a T–S Fuzzy Model: A TP Model Transformation-Based Approach, by P. Baranyi, *IEEE Transactions on Fuzzy Systems*, Vol. 30, No. 2, Feb 2022, pp. 345–356.

Digital Object Identifier: 10.1109/TFUZZ.2020.3038488

"The motivation behind 15 years of continuous development within the topic of the tensor product (TP) model transformation is that the greater the number of parameters or components of the Takagi–Sugeno (T–S) fuzzy model one can manipulate, the larger complexity reduction or control optimization one can achieve. This article proposes a radically new type of extension to the TP model transformation. While earlier variants of the TP model transformation focused on how the antecedent—consequent fuzzy set system of a

Digital Object Identifier 10.1109/MCI.2022.3180882
Date of current version: 19 July 2022

given T–S fuzzy model could be varied, this article, in contrast, focuses on how the number of inputs to a given T–S fuzzy model can be manipulated. The proposed extension is capable of changing the number of inputs or transforming the nonlinearity between the fuzzy rules and the input dimensions. These new features considerably increase the modeling power of the TP model transformation, allowing for further complexity reduction and more powerful control optimization to be achieved. This article provides two examples to show how the proposed extension can be used in a routine-like fashion."

IEEE Transactions on Evolutionary Computation

Adaptive Multifactorial Evolutionary Optimization for Multitask Reinforcement Learning, by A. D. Martinez, J. Del Ser, E. Osaba, and F. Herrera, *IEEE Transactions on Evolutionary Computation*, Vol. 26, No. 2, Apr 2022, pp. 233–247.

Digital Object Identifier: 10.1109/TEVC.2021.3083362

"Evolutionary computation has largely exhibited its potential to complement conventional learning algorithms in a variety of machine learning tasks, especially those related to unsupervised (clustering) and supervised learning. It has not been until lately when the computational efficiency of evolutionary solvers has been put in prospective for training reinforcement learning models. However, most studies framed so far within this context have considered environments and tasks conceived in isolation, without any exchange of knowledge among related tasks. In this manuscript we present A-MFEA-RL, an adaptive version of the well-known MFEA algorithm whose search and inheritance operators are tailored for multitask reinforcement learning environments. Specifically, our approach includes crossover and inheritance mechanisms for refining the exchange of genetic material, which rely on the multilayered structure of modern deep-learning-based reinforcement learning models. In order to assess the performance of the proposed approach, we design an extensive experimental setup comprising multiple reinforcement learning environments of varying levels of complexity, over which the performance of A-MFEA-RL is compared to that furnished by alternative nonevolutionary multitask reinforcement learning approaches. As concluded from the discussion of the obtained results, A-MFEA-RL not only achieves competitive success rates over the simultaneously addressed tasks, but also fosters the exchange of knowledge among tasks that could be intuitively expected to keep a degree of synergistic relationship."

IEEE Transactions on Games

Training a Gaming Agent on Brainwaves, by B. Francisco, M. Juan, N. Natalia, V. José, R. Rodrigo, and S. J. Miguel, *IEEE Transactions on Games*, Vol. 14, No. 1, Mar 2022, pp. 85–92.

Digital Object Identifier: 10.1109/TG.2020.3042900

"Error-related potentials (ErrPs) are a particular type of event-related potential elicited by a person attending a recognizable error. These electroencephalographic signals can be used to train a gaming agent by a reinforcement learning algorithm to learn an optimal policy. The experimental process consists of an observational human critic (OHC) observing a simple game scenario while their brain signals are captured. The game consists of a grid, where a blue spot has to reach a desired target in the fewest amount of steps. Results show that there is an effective transfer of information and that the agent successfully learns to solve the game efficiently, from the initial 97 steps on average required to reach the target to the optimal number of eight steps. Our results are expressed in threefold: the mechanics of a simple grid-based game that can elicit the ErrP signal component; the verification that the reward function only penalizes wrong steps, which means that type II error (not properly identifying a wrong movement) does not affect significantly the agent learning process; collaborative rewards from multiple OHCs can be used to train the algorithm effectively and can compensate low classification accuracies and a limited scope of transfer learning schemes."

IEEE Transactions on Cognitive and Developmental Systems

Behavior Decision of Mobile Robot With a Neurophysiologically Motivated Reinforcement Learning Model, by D. Wang, S. Chen, Y. Hu, L. Liu, and H. Wang, *IEEE Transactions on Cognitive and Developmental Systems*, Vol. 14, No. 1, Mar 2022, pp. 219–233.

Digital Object Identifier: 10.1109/TCDS.2020.3035778

"Online model-free reinforcement learning (RL) approaches play a crucial role in coping with the real-world applications, such as the behavioral decision making in robotics. How to balance the exploration and exploitation processes is a central problem in RL. A balanced ratio of exploration/exploitation has a great influence on the total learning time and the quality of the learned strategy. Therefore, various action selection policies have been presented to obtain a balance between the exploration and exploitation procedures. However, these approaches are rarely, automatically, and dynamically regulated to the environment variations. One of the most amazing self-adaptation mechanisms in animals is their capacity to dynamically switch between exploration and exploitation strategies. This article proposes a novel neurophysiologically motivated model which simulates the role of medial prefrontal cortex (MPFC) and lateral prefrontal cortex (LPFC) in behavior decision. The sensory input is transmitted to the MPFC, then the ventral tegmental area (VTA) receives a reward and calculates a dopaminergic reinforcement signal, and the feedback categorization neurons in anterior cingulate cortex (ACC) calculate the vigilance according to the dopaminergic reinforcement signal. Then the vigilance is transformed to LPFC to regulate the

exploration rate, finally the exploration rate is transmitted to thalamus to calculate the corresponding action probability. This action selection mechanism is introduced to the actor–critic model of the basal ganglia, combining with the cerebellum model based on the developmental network to construct a new hybrid neuromodulatory model to select the action of the agent. Both the simulation comparison with other four traditional action selection policies and the physical experiment results demonstrate the potential of the proposed neuromodulatory model in action selection."

IEEE Transactions on Emerging Topics in Computational Intelligence

A Survey of Embodied AI: From Simulators to Research Tracks, by J. Duan, S. Yu, H. L. Tan, H. Zhu, and C. Tan, *IEEE Transactions on Emerging Topics in Computational Intelligence*, Vol. 6, No. 2, Apr 2022, pp. 230–244.

Digital Object Identifier: 10.1109/TETCI.2022.3141105

"There has been an emerging paradigm shift from the era of "internet AI" to "embodied AI," where AI algorithms and agents no longer learn from datasets of images, videos or text curated primarily from the internet. Instead, they learn through interactions with their environments from an egocentric perception similar to humans. Consequently, there has been substantial growth in the demand for embodied AI simulators to support various embodied AI research tasks. This growing interest in embodied AI is beneficial to the greater pursuit of Artificial General Intelligence (AGI), but there has not been a contemporary and comprehensive survey of this field. This paper aims to provide an encyclopedic survey for the field of embodied AI, from its simulators to its research. By evaluating nine current embodied AI simulators with our proposed seven features, this paper aims to understand the simulators in their provision for use in embodied AI research and their limitations. Lastly, this paper surveys the three main research tasks in embodied AI—visual exploration, visual navigation and embodied question answering (QA), covering the state-of-the-art approaches, evaluation metrics and datasets. Finally, with the new insights revealed through surveying the field, the paper will provide suggestions for simulator-for-task selections and recommendations for the future directions of the field."

IEEE Transactions on Artificial Intelligence

Challenges and Countermeasures for Adversarial Attacks on Deep Reinforcement Learning, by I. Ilahi, M. Usama, J. Qadir, M. U. Janjua, A. Al-Fuqaha, D. T. Hoang, and D. Niyato, *IEEE Transactions on Artificial Intelligence*, Vol. 3, No. 2, Apr 2022, pp 90–109.

Digital Object Identifier: 10.1109/TAI.2021.3111139

"Deep reinforcement learning (DRL) has numerous applications in the real world, thanks to its ability to achieve high performance in a range of environments with little manual oversight. Despite its great advantages, DRL is susceptible to adversarial attacks, which precludes its use in real-life critical systems and applications (e.g., smart grids, traffic controls, and autonomous vehicles) unless its vulnerabilities are addressed and mitigated. To address this problem, we provide a comprehensive survey that discusses emerging attacks on DRL-based systems and the potential countermeasures to defend against these attacks. We first review the fundamental background on DRL and present emerging adversarial attacks on machine learning techniques. We then investigate the vulnerabilities that an adversary can exploit to attack DRL along with state-of-the-art countermeasures to prevent such attacks. Finally, we highlight open issues and research challenges for developing solutions to deal with attacks on DRL-based intelligent systems."

©SHUTTERSTOCK.COM/THAPANA_STUDIO

Privacy-Preserving Deep Learning With Homomorphic Encryption: An Introduction

Alessandro Falcetta and Manuel Roveri
Politecnico di Milano, ITALY

Abstract—Privacy-preserving deep learning with homomorphic encryption (HE) is a novel and promising research area aimed at designing deep learning solutions that operate while guaranteeing the privacy of user data. Designing privacy-preserving deep learning solutions requires one to completely rethink and redesign deep learning models and algorithms to match the severe technological and algorithmic constraints of HE. This paper provides an introduction to this complex research area as well as a methodology for designing privacy-preserving convolutional neural networks (CNNs). This methodology was applied to the design of a privacy-preserving version of the well-known LeNet-1 CNN, which was successfully operated on two benchmark datasets for image classification. Furthermore, this paper details and comments on the research challenges and software resources available for privacy-preserving deep learning with HE.

I. Introduction

Today's world is characterized by information abundance [1]. Thousands of exabytes of data are generated every day [2] by Internet-of-Things systems, mobile devices, social media, and industrial machinery. To extract value from these data, intelligent "data-processing" services have increased in number in recent years, which are based

Digital Object Identifier 10.1109/MCI.2022.3180883
Date of current version: 19 July 2022

Corresponding author: Alessandro Falcetta (e-mail: alessandro.falcetta@polimi.it).

1556-603X/22©2022IEEE

on machine and deep learning and operate on the cloud or in mobile apps [3]. Unfortunately, the processing of data acquired by users, companies, or stakeholders by third-party software services may severely impact privacy when sensitive data are involved (e.g., medical diagnoses, political or personal opinions, and confidential information) [4]. The need to combine privacy with intelligent services sheds light on one of the most relevant scientific and technological challenges of the coming years: *How can software services and mobile apps that provide intelligent functionalities (through machine and deep learning solutions) be designed while guaranteeing the privacy of user data*? This is a crucial question that research has begun to address from several perspectives, including scientific [5], technological [6], [7], and legislative [8]. Table I presents a comparison of the main approaches provided in the literature for integrating privacy constraints with intelligent processing abilities.

Homomorphic encryption schemes represent a special type of encryption that allows (a set of) operations to be performed on encrypted data.

Interestingly, among these families of solutions, *homomorphic encryption* (HE) is the only one that guarantees both the ability to process encrypted data as well as to operate without requiring multiple rounds of client-server computation/communication. HE schemes represent a special type of encryption that allows (a set of) operations to be performed on encrypted data. Specifically [9], an encryption function E and its decryption function D are homomorphic w.r.t. a class of functions $\mathcal{F}$ if, for any function $f \in \mathcal{F}$, one can construct a function g such that $f(m) = D(g(E(m)))$ for a set of input m.

Due to HE's ability to perform operations on encrypted data without multiple rounds of client-server communications, it is particularly suitable for consideration in the "as-a-service" computing paradigm, which requires high standards of privacy and data confidentiality. Indeed, integrating HE with machine and deep learning solutions could lead, for example, to the design of a cloud-based diagnosis system that is able to process X-ray images previously encrypted by a patient. The encrypted results (e.g., an index measuring the presence of potentially critical health threats) would be sent back to the patient, who would be the only one able to decrypt them.

Unfortunately, this ability comes at the expense of three drawbacks: *First*, only a subset of operations (mainly addition and multiplication) is allowed in most of HE-based processing systems; *second*, the length of the processing pipeline (i.e., the amount and type of operations to be executed) is restricted; and *third*, the memory and computational demand of HE-based processing systems are much higher than those of traditional systems.

These three drawbacks are particularly relevant in a scenario where deep learning solutions are considered, since deep learning models are typically characterized by a long pipeline of processing layers that comprises various types of nonlinear operations. For this reason, deep learning models and solutions must be completely redesigned and redeveloped to consider the constraints of HE schemes. Only a few studies have proposed addressing this issue with effective solutions in highly specific fields [10], and a general approach to HE for deep learning is still missing. Therefore, the aim of this study was twofold:

- To introduce HE for machine and deep learning by describing HE encoding/encryption mechanisms and the operations on plaintexts/ciphertexts;
- To provide a methodology for the step-by-step design of privacy-preserving convolutional neural networks (CNNs) based on HE. The goal of this methodology is to trace the path in the design of HE-based machine and deep learning solutions for supporting privacy-preserving intelligent processing in cloud-based services or mobile apps.

To achieve these aims, this study complemented theory with examples and code by applying the proposed methodology to the design of a privacy-preserving version of the well-known LeNet-1 CNN [11]. Experimental results are presented regarding the effectiveness and efficiency of the privacy-preserving LeNet-1 on two benchmark datasets for image classification. Furthermore, the research challenges and the software resources available for the design of privacy-preserving deep learning solutions are detailed and commented on. In addition, all of the codes used in this study have been made available to the scientific community as a public repository.[1]

The remainder of this paper is organized as follows. Section II introduces the Brakerski–Fan–Vercauteren (BFV) scheme for HE together with the encoding/decoding mechanisms and privacy-preserving operations. Section III introduces the proposed methodology for the design of privacy-preserving CNNs with HE, and then Section IV details the application of this methodology to the well-known LeNet-1 CNN. Finally, the research challenges and software resources available for privacy-preserving deep learning are presented in Sections V and VI, respectively, before the conclusions of the study are drawn in Section VII.

II. Homomorphic Encryption: The BFV Scheme

This section illustrates the main characteristics of HE schemes and provides concrete examples of its algebraic peculiarities. An HE scheme is an encryption scheme that supports the

[1]https://github.com/AlexMV12/Introduction-to-BFV-HE-ML

TABLE I Comparison of methodologies for privacy-preserving machine learning.

	Ability to Process Encrypted Data	Processing Without the Need for Multiple Rounds of Communication
Homomorphic encryption	Yes	Yes
Multi-party computation	Yes	No
Group-based anonymity	No	Yes
Differential privacy	No	Yes

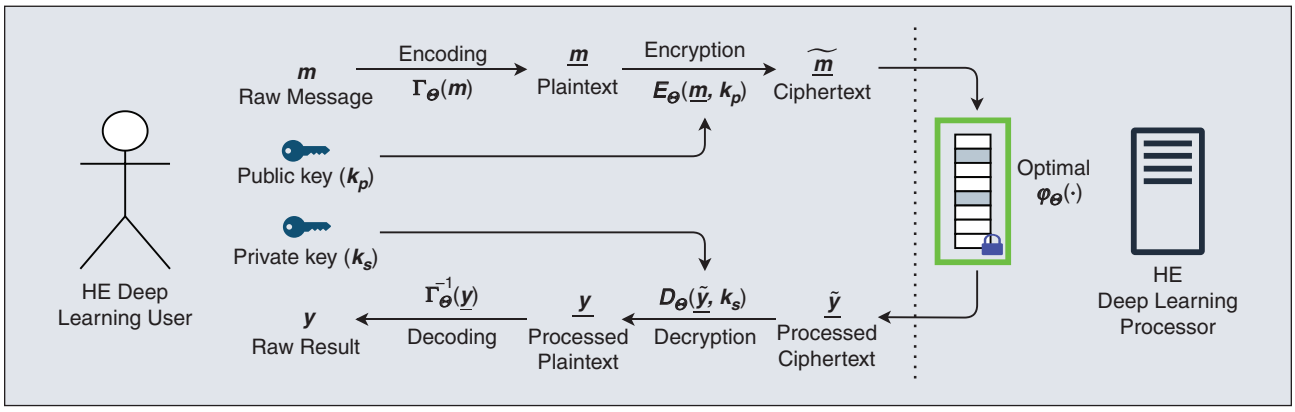


FIGURE 1 Machine and deep learning processing chain based on the BFV homomorphic encryption scheme.

computation of a set of operations directly on encrypted data. This goal is achieved thanks to the HE scheme's ability to maintain the algebraic structure of the data during the encrypted processing. In other words, HE allows a party to compute operations between ciphertexts, guaranteeing that the obtained result, when decrypted, will be equal (under certain assumptions, which are detailed as follows) to that obtained by computing the same operations between the corresponding plaintexts.

HE schemes can generally be classified into four categories, which are characterized by increasing complexity in terms of the number and type of operations. First, *partially HE schemes* are HE schemes that can support the homomorphic computation of only one class of operations. An example of a partially HE scheme is Rivest–Shamir–Adleman (RSA) [12].

TABLE II Summary of the notations used in this paper.

NOTATION	MEANING
m	Raw message
n	Polynomial modulus degree
p	Plaintext coefficient modulus
q	Ciphertext coefficient modulus
Θ	Encryption parameters (n, p, q)
Φ_n	Cyclotomic polynomial (x^n+1)
$R_p = \mathbb{Z}_p[x]/\Phi_n(x)$	Plaintext polynomial ring
$R_q = \mathbb{Z}_q[x]/\Phi_n(x)$	Ciphertext polynomial ring
$\underline{m}$	Encoded message (plaintext)
$\tilde{\underline{m}}$	Encrypted message (ciphertext)
$\underline{m} = \Gamma_\Theta(m)$	Encoding
$m = \Gamma^{-1}_\Theta(\underline{m})$	Decoding
k_s, k_p	Secret and public keys
$\tilde{\underline{m}} = E_\Theta(\underline{m}, k_p)$	Encryption
$\underline{m} = D_\Theta(\tilde{\underline{m}}, k_s)$	Decryption
$\lfloor \cdots \rfloor$	Floor operator
$\lceil \cdots \rfloor$	Round operator
$[\cdots]_{\Phi_n p}$	Modulo x^n+1, modulo p

Second, *somewhat HE schemes* support the homomorphic computation of an unbounded number of additions and a single multiplication. A notable example in this category is the Boneh–Goh–Nissim (BGN) scheme [13]. Third, *leveled HE schemes* are schemes that support the homomorphic computation of a predetermined number of additions and multiplications. The BFV scheme [14] (which is the reference HE scheme considered in this paper) and the Cheon–Kim–Kim–Song (CKKS) scheme [15] belong to this category. Finally, *fully HE schemes* support the homomorphic computation of an unbounded number of operations, often binary ones (AND, NOT…). While the HE schemes in this category provide the greatest flexibility in terms of processing abilities, configuring and managing them is very difficult. An example of a fully HE scheme is Torus–Fully–HE (TFHE) [16]. As previously mentioned, this paper focuses on the BFV scheme [14], which is a popular leveled HE scheme based on the ring learning with errors (RLWE) problem [17].

A deep learning solution that implements the BFV scheme is summarized in Figure 1, where the raw message m (i.e., the message to be processed in an encrypted manner) is a vector of numbers representing, for instance, an image or an audio clip according to the considered deep learning task. The raw message m is initially transformed into an encoded message $\underline{m}$ (called *plaintext*) by means of the *encoding* step $\Gamma_\Theta(\cdot)$ which transforms every number in m into a BFV polynomial. The plaintext $\underline{m}$ is then encrypted into the encrypted message $\tilde{\underline{m}}$ (called *ciphertext*) by means of the *encryption* step $E_\Theta(\cdot)$. Encoding and encryption, detailed in Sections II-B and II-C, respectively, depend on the BFV parameters Θ, which are detailed in Section II-A.

The core of the BFV scheme is its ability to support the computation of additions and multiplications between ciphertexts and ciphertexts as well as between ciphertexts and plaintexts. Additions and multiplications in the BFV scheme are detailed in Section II-D. Then, in Section II-E, the "noise budget" is described, which is an integer value that measures the number and type of operations that can be executed while guaranteeing that input data are correctly processed. Table II summarizes notations that appear throughout this paper.

A. Encryption Parameters of the BFV Scheme

The BFV scheme relies on the following set $\Theta = [n, p, q]$ of encryption parameters:

- n: Polynomial modulus degree;
- p: Plaintext coefficient modulus;
- q: Ciphertext coefficient modulus.

The core of the BFV scheme is its ability to support the computation of additions and multiplications between ciphertexts and ciphertexts as well as between ciphertexts and plaintexts.

As detailed in Section II-B, ciphertexts and plaintexts are represented by polynomials in the BFV scheme; these parameters define the order of the BFV polynomials and the range of values for their coefficients. Specifically, the parameter n must be a positive power of 2 and represents the degree of the cyclotomic polynomial $\Phi_n(x)$. In particular, the polynomial $\Phi_n(x) = x^n + 1$ represents the polynomial modulus. The plaintext modulus p is a positive integer that represents the module of the coefficients of the polynomial ring $R_p = \mathbb{Z}_p[x]/\Phi_n(x)$ (on which the RLWE problem is based). Finally, the parameter q is a large positive integer (larger than p) that results from the product of distinct prime numbers. It represents the modulo of the coefficients of the polynomial ring in the ciphertext space.

The setting of these parameters as well as their effect on the *noise budget* are described and commented on in the rest of the section.

B. Encoding and Decoding

Throughout the remainder of this section, raw messages ms are considered unsigned integers. Each number can be transformed into a BFV polynomial by means of the encoding step. Formally, in the BFV scheme the encoding step

$$\underline{m} = \Gamma_\Theta(m)$$

aims to transform m into an n-degree polynomial defined as follows:

$$\underline{m} = c_{n-1}x^{n-1} + \cdots + c_1 x^1 + c_0 \tag{1}$$

whose coefficients c_is are modulus p, that is, $c_i \in \mathcal{N} \setminus p$, $i = 0, \ldots, n-1$, where n and p are the polynomial modulus degree and the plaintext coefficient modulus, respectively, as defined in Section II-A. Several methods for encoding numbers into polynomials are available in the literature (e.g., integer encoding and fractional encoding [18]); however, this section focuses on encoding based on binary representation [19], which is a widely used encoding mechanism for natural numbers. The basis of the method is the ability to initially encode m into an n-bit binary representation. These n bits are considered the n coefficients $[c_{n-1}, \ldots, c_0]$ of the n-degree polynomial defined in Eq. (1). Hence, $c_i = \{0, 1\}$ with $i = 0, \ldots, n-1$.

Noteworthily, the corresponding *decoding* step $\Gamma_\Theta^{-1}(\underline{m})$ based on binary representation simply refers to the evaluation of the polynomial $\underline{m}$ for $x = 2$:

$$m = \Gamma_\Theta^{-1}(\underline{m}) = \underline{m}(2).$$

For example, consider a BFV scheme with the parameters $n = 16$, $p = 7$ and $q = 874$ and assume that one wishes to encode the following two raw messages: $m_1 = 7$ and $m_2 = 2$. The binary representation of $m_1 = 7$ over $n = 16$ bits is $[0000000000000111]$; hence the corresponding plaintext becomes

$$\underline{m_1} = \Gamma_\Theta(m_1) = x^2 + x + 1.$$

Similarly, when considering $m_2 = 2$, the corresponding plaintext is

$$\underline{m_2} = \Gamma_\Theta(m_2) = x.$$

C. Encryption and Decryption

In the BFV scheme, a plaintext $\underline{m}$ is transformed into a ciphertext $\tilde{\underline{m}}$ by means of the *encryption* step $E_\Theta(\underline{m}, k_p)$, where k_p is the public key. The corresponding *decryption* step $D_\Theta(\tilde{\underline{m}}, k_s)$ transforms the ciphertext $\tilde{\underline{m}}$ into a plaintext $\underline{m}$ with the private key k_s.

The public and private keys, which are crucial for the encryption and decryption steps of the BFV scheme, are generated by the user before the encryption phase. Specifically, the secret key k_s corresponds to an n-degree polynomial whose coefficients are randomly selected in the set $\{-1, 0, 1\}$. Given k_s, the public key k_p is a couple (k_{p0}, k_{p1}) of n-degree polynomials, which are defined as follows:

$$k_p = (k_{p0}, k_{p1}) = ([-(ak_s + e)]_{\Phi_n, q}, a) \tag{2}$$

where a is an n-degree polynomial whose coefficients are randomly selected in the set $\{0, \ldots, q-1\}$ and e is an n-degree polynomial whose coefficients are randomly selected from a discrete and bounded Gaussian distribution $\mathcal{N}(\mu = 0, \sigma = 3.2)$ over the integers [20].

The *encryption* step

$$\tilde{\underline{m}} = E_\Theta(\underline{m}, k_p)$$

aims to transform the plaintext $\underline{m}$ into the ciphertext $\tilde{\underline{m}}$, which is a couple $(\widetilde{m_0}, \widetilde{m_1})$ of n-degree polynomials defined as follows:

$$\begin{aligned}\tilde{\underline{m}} &= (\widetilde{m_0}, \widetilde{m_1}) = \\ &= \left(\left[k_{p0}u + e_1 + \left\lfloor \frac{q}{p} \right\rfloor \cdot m\right]_{\Phi_n, q}, [k_{p1}u + e_2]_{\Phi_n, q}\right)\end{aligned} \tag{3}$$

where e_1, e_2 are n-degree polynomials computed as e, while u is an n-degree polynomial computed as k_s. Notably, in Eq. (3), the sum operator "+" refers to the sum between polynomials, whereas the

operator "·" represents the pointwise multiplication between a scalar value and a polynomial (i.e., all of the coefficients of $\underline{m}$ are multiplied by the factor $\lfloor q/p \rfloor$, which represents the floor of the division between q and p). Note that this factor is larger than one since q is larger than p. The two polynomials representing $\tilde{\underline{m}}$ are of order n and with coefficients that are modulus q.

Some noise is injected into the ciphertext by e_1, e_2, and u. In particular, u is a "mask" that actually hides the message in the ciphertext. Notably, these noise terms are randomly selected every time the encryption step is activated, and thus, they are responsible for guaranteeing the probabilistic encryption ability of the encryption scheme, which is a relevant property from a security point of view [21].

The decryption step of a ciphertext $\tilde{\underline{m}}$ operates as follows [14], [19]:

$$D_\Theta(\tilde{\underline{m}}, k_s) = \underline{m} = \left[\left[\frac{p}{q}[\widetilde{\underline{m_0}} + \widetilde{\underline{m_1}} k_s]_{\Phi_n, q}\right]\right]_p \tag{4}$$

where $\lfloor \cdot \rceil$ is the round operation. In the decryption phase, $[\widetilde{\underline{m_0}} + \widetilde{\underline{m_1}} k_s]_{\Phi_n, q}$ is scaled by the factor p/q. All of the coefficients are modulus p (after being rounded).

For the sake of clarity, the role of the secret key in the decryption step requires further elaboration. Expanding Eq. (3) w.r.t. the public key k_s leads to:

$$\tilde{\underline{m}} = \left(\left[-ak_s u - eu + e_1 + \left\lfloor \frac{q}{p} \right\rfloor \cdot \underline{m}\right]_{\Phi_n, q}, [au + e_2]_{\Phi_n, q}\right).$$

The encoded message m, multiplied by $\lfloor q/p \rfloor$, is included in the first term of the ciphertext, suitably hidden by a mask $(-ak_s u)$, and perturbed by noise $(-eu + e_1)$. The mask is also present in the second term (i.e., au) of the ciphertext together with the noise (e_2). The core of the decryption phase resides in the ability of the user to encrypt $\underline{m}$ by means of k_s and by the fact that k_s can be multiplied for the second term $\widetilde{\underline{m_1}}$ of the ciphertext. Intuitively, multiplying $\widetilde{\underline{m_1}}$ with k_s and summing with $\widetilde{\underline{m_0}}$ removes the mask u from the raw message $\underline{m}$, as long as the error terms—accumulated during the pipeline of homomorphic operations—are not too big. An example of this encryption/decryption phase is presented in Section II-F.

D. Homomorphic Operations in the BFV Scheme

The BFV scheme allows the computation of additions and multiplications between ciphertexts, between ciphertexts and plaintexts, and between plaintexts. The addition and multiplications in the BFV scheme are as follows:

- ❏ "modulus $x^n + 1$", where the polynomial that is the outcome of the operation is modulus the cyclotomic polynomial $\Phi_n(x) = x^n + 1$;
- ❏ "coefficient modulus" p;

where the operands are plaintexts;

- ❏ "modulus $x^n + 1$", where the polynomial that is the outcome of the operation is modulus the cyclotomic polynomial $\Phi_n(x) = x^n + 1$;
- ❏ "coefficient modulus" q;

where at least one of the two operands is a ciphertext.

Now, this example continues with $m_1 = 7$ and $m_2 = 2$ considering the addition and multiplication of these two plaintexts in the BFV scheme. In particular, the addition of $m_1 + m_2$ becomes, in polynomial form,

$$[\underline{m_1} + \underline{m_2}]_{\Phi_{16}, 7} = x^2 + 2x + 1,$$

while the corresponding decoding of $\underline{m_1} + \underline{m_2}$ becomes $\Gamma_\Theta^{-1}(\underline{m_1} + \underline{m_2}) = 9$. Similarly, the multiplication of m_1 and m_2 becomes

$$[\underline{m_1} * \underline{m_2}]_{\Phi_{16}, 7} = x^3 + x^2 + x,$$

while the decoding of $\underline{m_1} * \underline{m_2}$ leads to $\Gamma_\Theta^{-1}(\underline{m_1} * \underline{m_2}) = 14$.

Note that, with modulus $x^n + 1$ and p, the following two operations

$$\sum_{i=1}^{7} \underline{m_1}, \tag{5}$$

$$\prod_{i=1}^{16} \underline{m_2} \tag{6}$$

introduce an "overflow" in the processing, thereby leading to incorrect results. In fact, when one of the coefficients of the polynomial becomes equal to or larger than p at the end of the processing, an incorrect result will be obtained due to the modulo p operation. In particular, in Eq. (5), all of the coefficients of the polynomial become 7 at the end of the processing. Hence, when the modulo 7 operation is applied, the coefficients become 0 leading to the following incorrect result:

$$[7x^2 + 7x + 7]_7 = 0x^2 + 0x + 0.$$

By contrast, in Eq. (6), the problem is related to the modulo Φ_n operation performed on the polynomial. Indeed, the degree of the polynomial becomes 16 during the last multiplication, leading to the following loss of information:

$$[x^{16}]_{\Phi_{16}} = -1.$$

When an overflow occurs in the processing, the final decrypted value will differ from the correct one. The overflow issue, described here on additions and multiplications between two plaintexts, also affects the operations that comprise ciphertexts, which are described as follows.

The addition between two ciphertexts (e.g., $\widetilde{\underline{m_1}}$ and $\widetilde{\underline{m_2}}$) is as simple as computing their element-wise sum (recall that a ciphertext is a couple of polynomials):

$$\widetilde{\underline{m_1}} + \widetilde{\underline{m_2}} = \left([\widetilde{\underline{m_{1_0}}} + \widetilde{\underline{m_{2_0}}}]_{\Phi_n, q}, [\widetilde{\underline{m_{1_1}}} + \widetilde{\underline{m_{2_1}}}]_{\Phi_n, q}\right). \tag{7}$$

By contrast, the multiplication between two ciphertexts produces a three-term ciphertext:

$$\widetilde{\underline{m_1}} * \widetilde{\underline{m_2}} = \left(\left[\left[\frac{p}{q}\ \widetilde{\underline{m_{1_0}}}\,\widetilde{\underline{m_{2_0}}}\right]\right]_{\Phi_n,q},\right.$$
$$\left[\left[\frac{p}{q}\left(\widetilde{\underline{m_{1_0}}}\,\widetilde{\underline{m_{2_1}}} + \widetilde{\underline{m_{1_1}}}\,\widetilde{\underline{m_{2_0}}}\right)\right]\right]_{\Phi_n,q},$$
$$\left.\left[\left[\frac{p}{q}\,\widetilde{\underline{m_{1_1}}}\,\widetilde{\underline{m_{2_1}}}\right]\right]_{\Phi_n,q}\right). \tag{8}$$

Having a three-term ciphertext is not a problem, given that all of the procedures (decryption, addition, and multiplication) can be modified to work on ciphertexts of arbitrary dimensions. However, the larger the ciphertexts, the larger their memory and computational footprint. To address these problems, the BFV scheme supports an operation called *relinearization* [14] which receives as input a ciphertext with size k and returns a ciphertext with size $k-1, 2$ the minimum dimension allowed. Although relinearization slightly increases the noise in the ciphertext and requires some additional keys, relinearizing the ciphertext after every ciphertext–ciphertext multiplication is generally recommended.

E. Noise Budget

As detailed in Eq. (3), the BFV scheme (similar to other HE schemes presented in the literature) injects noise into ciphertexts during the encryption step. This is necessary to guarantee the probabilistic encryption property of the BFV scheme since encrypting the same plaintext through two different activations of the encryption step would lead to two different ciphertexts. The drawback is that, during homomorphic addition and multiplication on the ciphertext, noise is added as well as multiplied. This might lead to a critical scenario where, during processing, one of the coefficients of the ciphertext is rounded to an incorrect value during decryption (as in Eq. (4)), hence failing the decryption phase.

Noise handling is a crucial point in the BFV HE scheme. A correct evaluation of the number (and type) of operations allowed on the ciphertexts is crucial in the design of HE-based processing systems. This is exactly where the noise budget (NB) comes into play. While providing a formal definition of the NB is outside of the scope of this paper (see [19] for details), one can intuitively define it as an indicator of the number of operations that can be performed on a ciphertext before its decryption will fail.

Interestingly, the NB is a property of a ciphertext that varies during the processing pipeline. It is measured as a positive integer and depends on the parameters Θ of the BFV scheme. The NB is initially allocated to the ciphertext immediately after the encryption step. In general, increasing n will increase the amount of NB available in a freshly encrypted ciphertext. By contrast, increasing p and q will increase NB consumption during homomorphic operations. Identifying the values of Θ that guarantee the correct processing of the ciphertexts while reducing computation and memory complexity is crucial for HE-based systems, particularly for those that implement deep learning solutions. This aspect is addressed in Section III.

HE operations (additions and multiplications) applied on the ciphertext decrease the NB. As presented in Table III, the types of operations and operands significantly affect the amount of reduction to the NB. It is crucial to highlight that the decryption step on ciphertexts must be conducted before the NB reduces to 0; otherwise the decryption will fail. Computing the NB is highly complex, but specific HE tools are available for its estimation (see [19] for details).

Table IV depicts the effects of Θ, operations, and operands on NB consumption. As previously mentioned, the initial NB increases with n, thus increasing the number of subsequent operations that can be computed on the ciphertexts. On the other hand, increasing p will increase the NB consumption of the HE operations.

F. Example

This section presents an example application of the BFV scheme. The basic operations of the scheme are implemented in Python through the scientific computation library NumPy [22]. The code used in this example has been made available to the scientific community in the public repository specified in the Section I.

First, this study defines the encryption parameters Θ of the BFV as follows:

```
n = 16
p = 7
q = 124112
theta = (n, p, q)
```

TABLE III Noise Budget (NB) consumption, on the basis of the computed homomorphic operation.

OPERATION	NB CONSUMPTION
Ciphertext-ciphertext multiplication	High
Ciphertext-ciphertext addition	Medium
Ciphertext-plaintext multiplication	Low
Ciphertext-plaintext addition	Very low

TABLE IV Example of NB consumption, using two different configurations of $\Theta = (n, p)$. The value for q is set automatically according to the SEAL library [19].

	NOISE BUDGET	
CIPHERTEXT	Θ=(2048, 15162)	Θ=(4096, 151262)
Freshly encrypted $\widetilde{\underline{m_1}}$	30	81
Freshly encrypted $\widetilde{\underline{m_2}}$	30	81
$\widetilde{\underline{m_1}} + \widetilde{\underline{m_2}}$	29	80
$\widetilde{\underline{m_1}} * \widetilde{\underline{m_2}}$	5	52
$\widetilde{\underline{m_1}} + \underline{m_2}$	30	81
$\widetilde{\underline{m_1}} * \underline{m_2}$	29	81

To create polynomials the function POLY is used; it accepts a dictionary of coefficients, and returns the corresponding NumPy polynomial. NumPy polynomials implement the basic operations of polynomial additions, multiplications, and evaluation among others.

For instance, the cyclotomic polynomial Φ_{16}, which is used in the following operations, is generated as follows:

```
cyclotomic_poly = Poly({0: 1, 16: 1})
print(cyclotomic_poly)
----------------------------------------
X**16 + 1
```

The function GENERATE_KEYS generates a pair of public and secret keys (k_s, k_p) according to Eq. (2), given Θ.

```
ks, kp = generate_keys(theta)
print("Secret key: " + ks)
print("Public key: " + kp)
----------------------------------------

Secret key: X**15 - X**14 - X**13 - X**12 + X
    **11 + X**10 - X**9 - X**8 - X**7 + X**6 -
      X**5 + X**4 - X**3 - X**2 + 1

Public key: 61199X**15 + 48133X**14 +... +
    62895X**2 + 113586X + 92746, 80534X**15 +
    36864X**14 + ... + 27318X**2 + 35006X +
    72552
```

It is now possible to encode and encrypt the two values of $m_1 = 7$ and $m_2 = 2$. An example of addition and multiplication is provided as follows.

First, the two values must be encoded as specified in Section II-B. To achieve this, the function ENCODE_BINARY takes as input an unsigned integer and returns the corresponding polynomial encoded with binary encoding. The DECODE_BINARY function is defined accordingly.

```
m1 = encode_binary(7)
m2 = encode_binary(2)
print("m1: " + m1)
print("m2: " + m2)
----------------------------------------

m1: X**2 + X + 1
m2: X
```

After the values have been encoded in polynomial form, it is possible to proceed with the encryption. First, the ENCRYPT_POLY function creates a ciphertext $\underline{\tilde{m}}$ that starts from an encoded polynomial $\underline{m}$. To this end, random polynomials e_1, e_2, u are generated and the ciphertext is created following Eq. (3).

```
enc_m1 = encrypt_poly(m1, kp, theta)
enc_m2 = encrypt_poly(m2, kp, theta)
print("enc_m1: " + enc_m1)
print("enc_m2: " + enc_m2)
----------------------------------------

enc_m1: 74404X**15 + 9539X**14 + ... + 27250X
    **2 + 64856X + 51090, 121602X**15 + 100582
    X**14 + ... + 60289X**2 + 80948X + 13677

enc_m2: 14068X**15 + 6425X**14 + ... + 23691X
    **2 + 53327X + 52978, 100091X**15 + 85920X
    **14 + ... + 16044X**2 + 51146X + 117168
```

Notably, after the encryption, the coefficients of the involved polynomials are now modulo q, that is in the range $[0, q)$. The maximum degree of the polynomials is 15, as this work is being conducted in the modulo Φ_{16} space.

It is now possible to sum up the two ciphertexts. The function ADD_CIPHERTEXTS simply computes the element-wise sum of the two ciphertexts while applying the modulus operations, as specified in Eq. (7).

```
enc_sum = add_ciphertexts(enc_m1, enc_m2)
print("Sum ciphertext: " + enc_sum)
----------------------------------------

Sum ciphertext: 88472X**15 + 15964X**14 + ...
    + 50941X**2 + 118183X + 104068, 97581X**15
     + 62390X**14 + ... + 76333X**2 + 7982X +
    6733
```

Finally, the result of the addition can be decrypted. For this purpose, the function DECRYPT_POLY, implemented according to Eq. (4), can be used as follows:

```
decrypted = decrypt_poly(enc_sum, ks, theta)
decoded = decode_binary(decrypted)

print("Decrypted encoded result: " +
     decrypted)
print("Decoded result: " + decoded))
----------------------------------------

Decrypted encoded result: X**2 + 2X + 1
Decoded result: 9
```

The result is the same as the result that would have been obtained on plain data. The same holds for multiplications. To this end, the function MUL_CIPHERTEXTS is defined. Note that the function will return a three-terms ciphertext, as depicted in Eq. (8).

```
enc_prod = mul_ciphertexts(enc_m1, enc_m2)
print("Product ciphertext: " + enc_prod)
----------------------------------------

Product ciphertext: 4616X**15 + 69851X**14 +
    ... + 70424X**2 + 108145X + 97186, 41476X
    **15 + 15425X**14 + ... + 13366X**2 +
    108332X + 94209, 77259X**15 + 32461X**14 +
     ... + 49450X**2 + 5811X + 35650
```

Decryption can now be used as follows:

```
decrypted = decrypt_poly(enc_prod, ks, theta)
decoded = decode_binary(decrypted)

print("Decrypted encoded result: " +
    decrypted)
print("Decoded result: " + decoded))
----------------------------------------

Decrypted encoded result: X**3 + X**2 + X
Decoded result: 14
```

In this case, the homomorphic multiplication between the two ciphertexts also produces a correct result. This ends the first part of this study, which was the introduction to HE with examples. In the next section, HE will be used for designing privacy-preserving CNNs.

III. Designing Privacy-Preserving CNNs With Homomorphic Encryption: A Methodology

The aim of this section is to detail the methodology for designing privacy-preserving CNNs using HE, which is the second of the two main contributions of this study. Designing privacy-preserving deep learning solutions based on HE requires one to rethink and redesign deep learning solutions that consider the constraints on the type and number of operations that characterize the BFV scheme. Among the wide range of deep learning solutions, the present paper focuses on CNNs [11], which are the state-of-the-art solution in several application scenarios, such as object detection and sound recognition [23].

A CNN $F(\cdot)$ is a deep neural network with L processing layers $\eta_{\xi_l}^{(l)}$, each of which is characterized by the parameters ξ_l with $l = 1, \ldots, L$. In a privacy-preserving CNN based on HE, the processing layers must comprise only addition and multiplication, and the length L of the processing pipeline must guarantee that the NB is not exhausted during processing.

Figure 2 presents an overview of the proposed methodology for the design of privacy-preserving CNNs based on HE. Let $F(\cdot)$ be the CNN to be encoded with HE. The goal of the methodology is to design a privacy-preserving version $\varphi_\Theta(\cdot)$ of $F(\cdot)$ as well as to identify the configuration of the encryption parameters Θ which will guarantee that the NB does not reach 0 during processing. To achieve these goals, the methodology has three different steps: model approximation, model encoding, and model validation. These three steps are detailed in the following sections.

A. Model Approximation

The aim of the *model approximation* is to replace the processing layers of $F(\cdot)$ that do not comply with the BFV scheme with those that rely only on additions and multiplications. Doing so is crucial in a deep learning scenario where processing layers typically comprise division, square root as well as nonlinear activation functions. The output of this step is an approximated model $\varphi(\cdot)$ that comprises processing layers that are HE-compliant.

In more detail, the model approximation step receives as input a CNN $F(\cdot)$ and provides the corresponding approximated model $\varphi(\cdot)$ as output. In such a model each processing layer $\eta_{\xi_l}^{(l)}$ comprises only additions and multiplications, and hence, is compatible with the BFV scheme. After this approximation phase, the approximated CNN model must be retrained.

A summary of the possible approximations is presented in Fig. 3 and detailed in the following paragraphs.

1) Pooling Layers

Maximum pooling layers use the comparison operator, which is not available in the BFV scheme. To address this problem, various pooling algorithms can be used to replace the maximum pooling. The present authors suggest replacing it with the average pooling available in the BFV scheme since it only requires multiplication between the sum of ciphertexts and a fixed value, which is known a priori (i.e., $1/k_w \times k_h$, with k_w, k_h being the width and height of the pooling kernel, respectively).

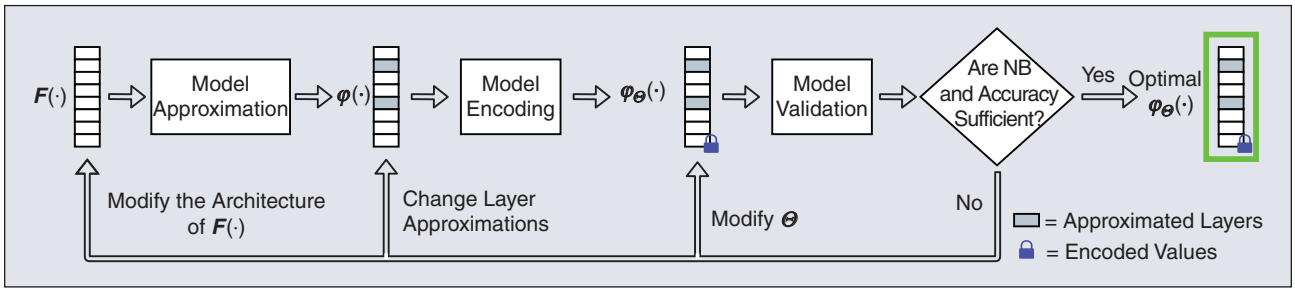


FIGURE 2 A methodology for the design of privacy-preserving CNNs based on HE.

Designing privacy-preserving deep learning solutions based on HE requires one to rethink and redesign deep learning solutions that consider the constraints on the type and number of operations that characterize the BFV scheme.

2) Normalization Layers

Normalization layers cannot be considered in the BFV scheme since it is impossible to compute the mean and standard deviation of encrypted data. By contrast, batch normalization layers are available given that they depend on the values of the data used for training. Such values are computed during the training and can be used during the processing of ciphertexts.

3) Activation Functions

The activation functions used in CNNs typically comprise nonlinear functions. The *ReLU* activation function, for instance, cannot be computed because it requires the use of the comparison operator. The same holds for the hyperbolic tangent *tanh* which involves division. This study suggests replacing these nonlinear activation functions with the square activation function $f(x) = x^2$. Such an approximation can be further refined using Taylor polynomial expansions. However, increasing the accuracy of the expansion (and thus using a larger number of polynomials) entails an increase in the number of operations (and thus an increased NB consumption).

B. Model Encoding

Once the model has been approximated, it can be encoded through the *model encoding* step, where an encoded approximated model $\varphi_\Theta(\cdot)$ is obtained, whose weights have been encoded according to the BFV scheme and the parameters Θ. Encrypted data can now be processed by $\varphi_\Theta(\cdot)$ to obtain the result of CNN processing. Notably, this result is still encrypted and only the owner of the secret key k_s will be able to decrypt it. The proper setting parameters Θ in HE processing remains an open research. Generally, the selection of the value of these parameters follows a "trial-and-error" approach (see the model validation step in Section III-C). Nevertheless, some guidelines for the setting of Θ are provided as follows.

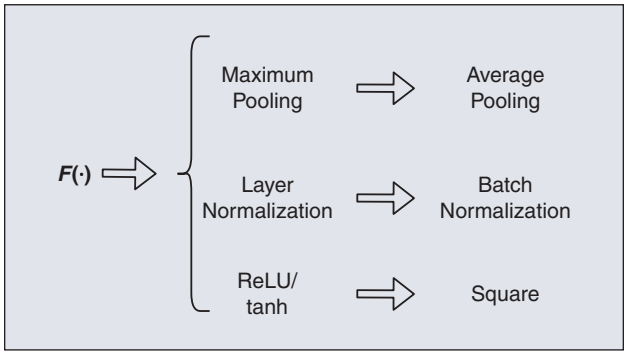


FIGURE 3 Possible approximations for typical CNN layers.

The most critical encryption parameter is n as it is a relevant parameter for the setting of the initial NB and the computational overhead of the encrypted processing. Generally, values of n smaller than 4096 are able to guarantee a NB sufficient only for very simple machine learning models (typically comprising two or three processing layers at most). From a methodological point of view, n is typically initially set to 4096 and then increased as described in Section III-C.

The parameter p affects NB consumption as well as the precision of the homomorphic operations (i.e., p affects the possibility that some coefficients of the decrypted polynomials are rounded to the incorrect value). Tuning p requires a trial-and-error process; typically, a value of p between 2^{16} and 2^{18} represents a good starting point for exploring the parameters described in Section III-C. The value of q is critical for the security of the scheme; it is suggested to rely on the helper function provided by SEAL [19] (see Section VI for details) to set q according to n and p.

The obtained $\Theta = (n, p, q)$ can be used to encode $\varphi(\cdot)$, thus obtaining the encoded deep learning model $\varphi_\Theta(\cdot)$.

C. Model Validation

Once the encoding step is completed, the *model validation* step is activated to evaluate the encoded model $\varphi_\Theta(\cdot)$ from two different perspectives. First, $\varphi_\Theta(\cdot)$ is evaluated to check whether the selected configuration Θ provides a sufficient NB in the processing of ciphertexts. Second, the loss in accuracy of $\varphi_\Theta(\cdot)$ w.r.t. $F(\cdot)$ is evaluated. To achieve both goals, a (possibly large) set of raw messages ms is processed by $\varphi_\Theta(\cdot)$ with the aim of measuring the NB of the final ciphertexts and evaluating the discrepancy between the accuracy of the encoded model $\varphi_\Theta(\cdot)$ and that of the plain model $F(\cdot)$. Typically, the problems associated with a loss of NB depend on an incorrect setting of Θ, whereas the discrepancies in the output between $\varphi_\Theta(\cdot)$ and $F(\cdot)$ could be associated with either the approximations of the processing layers introduced in the model approximation step or the fact that p and q are not large enough for the processing pipeline defined in the encoded approximated model $\varphi_\Theta(\cdot)$.

If the constraint on the NB is satisfied and the loss of accuracy is below a user-defined threshold (e.g., 1% or 5%), $\varphi_\Theta(\cdot)$ becomes the privacy-preserving version of $F(\cdot)$ to be considered. Conversely, when either the constraint on the NB is not satisfied (i.e., the NB of the ciphertexts decreases to 0 during the processing of $\varphi_\Theta(\cdot)$) or the loss in accuracy is larger than the threshold, the methodology suggests three different actions: update Θ, modify how layers in $F(\cdot)$ are approximated, or change the processing pipeline of $F(\cdot)$. These three actions are described in detail as follows.

First, the NB and loss of accuracy strictly depend on Θ. In particular, increasing the parameter n increases the initial NB but at the expense of a (potentially large) increase in computational overhead and the memory demand of $\varphi_\Theta(\cdot)$. Conversely, increasing p and q would reduce the loss in accuracy (by

increasing the precision of the processing) but at the expense of increased NB consumption by the HE operations.

Second, different model approximations could be considered for the processing layers that are not HE-compliant in $F(\cdot)$. Here, a trade-off must be carefully explored. Indeed, to reduce the loss of accuracy, a coarse-grain layer approximation could be replaced by a finer one (e.g., by using a higher degree of polynomial approximation); however, this would be at the expense of an increased number of operations to be performed for that layer (hence further reducing the NB). On the other hand, moving from a fine-grain layer approximation to a coarse-grain one could reduce NB consumption but possibly increase the loss of accuracy.

Third, if the previous two actions do not succeed in satisfying the constraints on NB and accuracy, a modified version $F'(\cdot)$ of $F(\cdot)$ can be designed. The aim is to reduce the number of operations to be conducted, such as by reducing the number of processing layers or simplifying the operations to be considered. Once $F'(\cdot)$ has been redesigned, the model approximation, encoding, and validation steps are newly activated to identify $\varphi_\Theta(\cdot)$.

Having detailed the proposed methodology, the next section will use it for the design of a privacy-preserving version of the well-known LeNet-1 CNN.

IV. Application to a Real-World CNN: Privacy-Preserving LeNet-1 With BFV

The aim of this section is to detail the application of the methodology proposed in this study to the LeNet-1 CNN, as well as provide numerical results to demonstrate its effectiveness and efficiency.

LeNet-1 is a simple yet effective CNN that was introduced by LeCun et al. [11]. Its processing pipeline $F(\cdot)$ is depicted in Fig. 4(a). LeNet-1 comprises $L = 5$ different processing layers: a convolutional layer with four kernels of size 5×5 and tanh activation; an average pooling layer with a kernel of size 2; a convolutional layer with 16 kernels of size 5×5 and tanh activation; an average pooling layer with a kernel of size 2; and a fully connected layer of size 192×10.

This study applied the methodology described in the previous section to LeNet-1 to design a privacy-preserving version $\varphi_\Theta(\cdot)$ compliant with the BFV scheme. The use of the methodology and the designed model $\varphi_\Theta(\cdot)$ are described in Section IV-A, and then the performance and accuracy of $\varphi_\Theta(\cdot)$ computed on the MNIST [24] and Fashion-MNIST [25] datasets are detailed in Section IV-B. Both datasets are composed of 70,000 28×28 gray-scale images (60,000 for training and 10,000 for testing), representing a 10-class classification problem, that is, 10 digits in MNIST and 10 fashion products in Fashion-MNIST.

A. Applying the Methodology to Design the Privacy-Preserving LeNet-1

This section presents the application of the methodology from Section III to the design of the privacy-preserving version $\varphi_\Theta(\cdot)$ of the LeNet-1 CNN $F(\cdot)$. During the model approximation step, this study replaced the tanh activation function of LeNet-1 (involving non-polynomial operations) with the square activation function. Moreover, during the model validation step, this study determined that the NB given by $n = 4096$ was not sufficient for conducting the processing of the encoded model on encrypted data. As mentioned in Section III-B, increasing n would have led to a higher NB but at the expense of a relevant increase in the computational overhead. Thus, this study explored the action of redesigning $F(\cdot)$ by removing the second tanh activation: this change in $F(\cdot)$ led us to consider $F'(\cdot)$, that is, a simplified version of the LeNet-1 CNN without the second tanh activation. Interestingly, $F'(\cdot)$ guaranteed a negligible loss in accuracy w.r.t. $F(\cdot)$ and led to a more effective definition of $\varphi_\Theta(\cdot)$ through the methodology. This aspect will be elaborated on in Section IV-B.

The final configuration of the parameters Θ was as follows:

$$\Theta = \begin{bmatrix} n = 4096 \\ p = 953983721 \\ q = 6.490334708969677435864154707968 * 10^{33} \end{bmatrix}.$$

In particular, the value for p was obtained with the trial-and-error procedure described in Section III-B, whereas the value of q was automatically computed by means of the specific SEAL function mentioned in Section III-B.

In summary, the outcome $\varphi_\Theta(\cdot)$ of the methodology representing the privacy-preserving version of LeNet-1 is depicted in Fig. 4(b). The processing pipeline of $\varphi_\Theta(\cdot)$ comprises five different layers: a convolutional layer with four kernels of size 5×5 and square activation; an average pooling layer with a kernel of size 2; a convolutional layer with 16 kernels of size 5×5; an

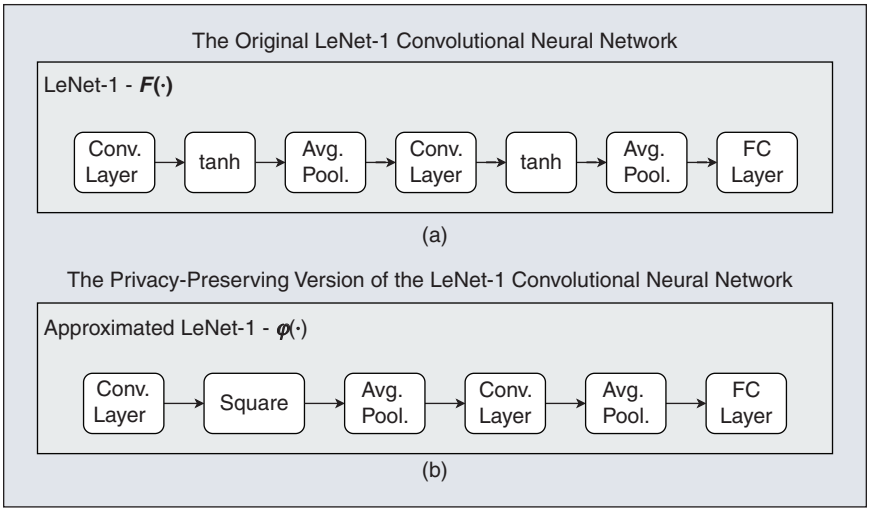


FIGURE 4 (a) The original LeNet-1 CNN and (b) its privacy-preserving version based on HE.

average pooling layer with a kernel of size 2; and a fully connected layer of size 192 × 10.

The Python code of the privacy-preserving LeNet-1 $\varphi_\Theta(\cdot)$ is available in the same repository cited in Section I.

B. Performance and Accuracy of the Privacy-Preserving LeNet-1

To demonstrate the effectiveness and efficiency of the designed privacy-preserving version of LeNet-1, we designed an experimental campaign aimed at measuring and comparing the accuracy, memory occupation (in MB) and computation time (in s) of the original LeNet-1 $F(\cdot)$, the simplified LeNet-1 $F'(\cdot)$ (without the second tanh activation), the approximated version $\varphi(\cdot)$ of $F'(\cdot)$, and the outcome $\varphi_\Theta(\cdot)$ of the methodology. The experiments were conducted on a machine with an Intel i7-4770K 64-bit CPU and 16 GB of RAM. The experimental results, computed on the testing images from the MNIST and Fashion-MNIST datasets, are detailed in Table V. Figure 5 graphically compares the accuracy, execution time, and memory occupation on the Fashion-MNIST dataset. The following three main comments can be made.

First, the simplified version $F'(\cdot)$ of the original LeNet-1 $F(\cdot)$ provided accuracies of 98.22% on MNIST and 86.23% on Fashion-MNIST. The drop in accuracy between $F(\cdot)$ and $F'(\cdot)$ was approximately 0.6% in both datasets. It was therefore crucial to verify whether the drop in accuracy induced by the use of $F'(\cdot)$ (instead of $F(\cdot)$) was acceptable.

Second, as expected, the model approximation step of the procedure, leading from $F'(\cdot)$ to $\varphi(\cdot)$ caused a loss of accuracy. This loss was negligible in the case of the MNIST dataset (0.04%), whereas it was more relevant in the case of the Fashion-MNIST dataset (0.94%). This study speculated that the removal of nonlinearities from LeNet-1 has a higher impact when more complex tasks are considered (e.g., the recognition of fashion products instead of simple digits). In addition, the accuracy of $\varphi(\cdot)$ was equal to that of $\varphi_\Theta(\cdot)$ meaning that the privacy-preserving model applied on encrypted images provided the same accuracy as that obtained on plain images. These results confirmed that, thanks to a correct choice of Θ and an accurate approximation of nonlinear layers, the discrepancy between the plain and encrypted results was negligible for the considered scenario.

Third, as expected, encrypted processing introduced a crucial overhead in terms of memory usage and computation time for the classification of a single image. Currently, HE libraries do not support parallelization; hence, GPU support for HE is only partially available with few examples present in the literature (e.g., [26]).

This ends the second main contribution of this study, which was the introduction of a methodology for designing privacy-preserving CNNs. The next section will discuss the main challenges to be addressed in this research field.

TABLE V Experimental results on the MNIST and Fashion-MNIST datasets. Memory occupation (MB) and computation time (s) refer to the processing of a single image. No parallelization of the code is considered.

Symbol	Model	Accuracy MNIST	Accuracy Fashion MNIST	Memory Occup.	Comp. Time
$F(\cdot)$	LeNet-1	98.76%	86.88%	7.6MB	0.001s
$F'(\cdot)$	LeNet-1 (single tanh)	98.22%	86.23%	6.5MB	0.0009s
$\varphi(\cdot)$	Approximated $F'(\cdot)$	98.18%	85.29%	6.5MB	0.0009s
$\varphi_\Theta(\cdot)$	$\varphi(\cdot)$ on encrypted data	98.18%	85.29%	780MB	138s

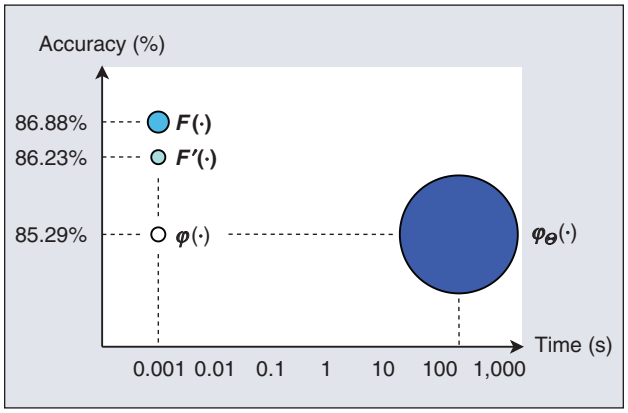


FIGURE 5 Time, accuracy, and memory occupation of the different models for the Fashion-MNIST dataset. The dimensions of the circles are representative of the memory occupation (in MB).

V. Homomorphic Encryption and Deep Learning: The Challenges

Designing privacy-preserving deep learning solutions is a novel research area with several open research challenges to address. This section, without aiming to be exhaustive, discussed three main challenges in this research area that will be relevant over the next few years:

- **Automatic parameter configuration:** The optimal parameter configuration Θ is currently selected using a trial-and-error approach. The challenge here lies in the study of optimization algorithms, theoretical solutions, and meta-learning algorithms (e.g., AutoML) to provide the optimal configuration of Θ given a processing pipeline $F(\cdot)$ and a reference dataset describing the problem to be addressed.
- **Privacy-preserving recurrent neural networks and transformers:** Currently, the privacy-preserving deep learning solutions in the literature have mostly focused on CNNs, whose transformation in HE-compliant models is easier than other deep learning solutions. One major challenge in this field is the design of privacy-preserving recurrent neural networks and transformers that are able to deal with data sequences. The major issue to be addressed here is the ability to manage the NB in processing pipelines where data are sequentially processed over time.
- **Training privacy-preserving models:** The literature regarding privacy-preserving machine and deep learning models with HE focuses on the inference of privacy-preserving

models. The next challenge to address is the training of machine and deep learning models directly on encrypted data. The main issue to be addressed here is to manage the NB consumption not only during the inference, but also during the training of the privacy-preserving model.

Having outlined the main challenges to be addressed in this research area, the next section presents a collection of useful resources for privacy-preserving deep learning with HE.

VI. Available Resources for Privacy-Preserving Deep Learning With Homomorphic Encryption

Two main frameworks for HE, which can be used to facilitate the design of privacy-preserving deep learning solutions, are available in the literature: SEAL and HElib. *SEAL* [19] is a Microsoft C++ library that implements the BFV and CKKS schemes. It offers helper functions for selecting the encryption parameters as well as provides support for basic HE operations (e.g., encrypting and decrypting values). Python users may refer to Pyfhel [27] and TenSEAL [28], which are Python wrappers for SEAL. The code used in the present study relies on SEAL and Pyfhel, while Torch [29] was used for the training of the plain-version of the CNNs. *HElib* [30] is a C++ library that implements the CKKS scheme, among others. HElib also includes optimization mechanisms for efficient homomorphic evaluation, focusing on the effective use of ciphertext packing techniques and Gentry–Halevi–Smart optimizations.

Concrete [31] is a Rust implementation of the TFHE scheme, while a few examples of software libraries specifically intended for HE-based machine and deep learning are available, such as PyCrCNN [32], nGraph-HE [9], and CHET [33].

VII. Conclusions

The aim of this study was to explore the promising but highly challenging research area of privacy-preserving deep learning based on HE. Specifically, the BFV scheme and its privacy-preserving operations were introduced both theoretically and algorithmically through Python code examples. A methodology for designing privacy-preserving CNNs was also proposed, which was applied to the design of a privacy-preserving version of the well-known LeNet-1. Experimental results on two datasets highlighted that it is possible to design privacy-preserving CNNs with HE, which are characterized by a negligible loss in accuracy (w.r.t. the original version) and relevant increases in memory and computational demand. Finally, this paper described the research challenges to be addressed in this field as well as the available software resources for privacy-preserving deep learning.

The path toward privacy-preserving deep learning with HE has now been traced. Over the next few years, great advances will be made in this direction.

Acknowledgment

This work was partially funded by the CATCH 4.0 project within the Italian *Programma Operativo Nazionale* (PON) "Imprese e competitività" FESR 2014/2020, and by Dhiria S.r.l., a spin-off of Politecnico di Milano.

References

[1] L. Cai and Y. Zhu, "The challenges of data quality and data quality assessment in the big data era," *Data Sci. J.*, vol. 14, 2015, doi: 10.5334/dsj-2015-002.

[2] S. Cass, "The age of the zettabyte cisco: The future of internet traffic is video [dataflow]," *IEEE Spectr.*, vol. 51, no. 3, pp. 68–68, Mar. 2014, doi: 10.1109/MSPEC.2014.6745894.

[3] Y. Yao, Z. Xiao, B. Wang, B. Viswanath, H. Zheng, and B. Y. Zhao, "Complexity vs. performance: empirical analysis of machine learning as a service," in *Proc. 2017 Internet Meas. Conf.*, 2017, pp. 384–397.

[4] E. P. Council of European Union, "Regulation (EU) no 2016/679, article 4(1)," 2016.

[5] B. C. Stahl and D. Wright, "Ethics and privacy in ai and big data: Implementing responsible research and innovation," *IEEE Security Privacy*, vol. 16, no. 3, pp. 26–33, May/Jun. 2018, doi: 10.1109/MSP.2018.2701164.

[6] H. C. Tanuwidjaja, R. Choi, S. Baek, and K. Kim, "Privacy-preserving deep learning on machine learning as a service—A comprehensive survey," *IEEE Access*, vol. 8, pp. 167,425–167,447, 2020.

[7] B. Pulido-Gaytan *et al.*, "Privacy-preserving neural networks with homomorphic encryption: Challenges and opportunities," *Peer-to-Peer Netw. Appl.*, vol. 14, no. 3, pp. 1666–1691, 2021, doi: 10.1007/s12083-021-01076-8.

[8] E. P. Council of European Union, "Regulation laying down harmonised rules on artificial intelligence (artificial intelligence act), com/2021/206," 2021.

[9] F. Boemer, Y. Lao, R. Cammarota, and C. Wierzynski, "nGraph-HE: A graph compiler for deep learning on homomorphically encrypted data," in *Proc. 16th ACM Int. Conf. Comput. Frontiers*, 2019, pp. 3–13.

[10] R. Gilad-Bachrach, N. Dowlin, K. Laine, K. Lauter, M. Naehrig, and J. Wernsing, "Cryptonets: Applying neural networks to encrypted data with high throughput and accuracy," in *Proc. Int. Conf. Mach. Learn.*, PMLR, 2016, pp. 201–210.

[11] Y. LeCun *et al.*, "Backpropagation applied to handwritten zip code recognition," *Neural Comput.*, vol. 1, no. 4, pp. 541–551, 1989, doi: 10.1162/neco.1989.1.4.541.

[12] L. Morris, "Analysis of partially and fully homomorphic encryption," *Rochester Inst. Technol.*, pp. 1–5, 2013.

[13] D. Boneh, E.-J. Goh, and K. Nissim, "Evaluating 2-DNF formulas on ciphertexts," in *Proc. Theory Cryptography Conf.*, Springer-Verlag, 2005, pp. 325–341.

[14] J. Fan and F. Vercauteren, "Somewhat practical fully homomorphic encryption." *IACR Cryptol. ePrint Arch.*, vol. 2012, p. 144, 2012.

[15] J. H. Cheon, A. Kim, M. Kim, and Y. Song, "Homomorphic encryption for arithmetic of approximate numbers," in *Proc. Adv. Cryptol. – ASIACRYPT 2017*, Cham, Switzerland: Springer Int. Publishing, 2017, pp. 409–437.

[16] I. Chillotti, N. Gama, M. Georgieva, and M. Izabachène, "TFHE: Fast fully homomorphic encryption over the torus," *J. Cryptol.*, vol. 33, no. 1, pp. 34–91, Jan. 2020, doi: 10.1007/s00145-019-09319-x.

[17] V. Lyubashevsky, C. Peikert, and O. Regev, "On ideal lattices and learning with errors over rings," in *Proc. Annu. Int. Conf. Theory Appl. Cryptographic Techn.*, Springer-Verlag, 2010, pp. 1–23.

[18] S. S. Sathya, P. Vepakomma, R. Raskar, R. Ramachandra, and S. Bhattacharya, "A review of homomorphic encryption libraries for secure computation," 2018, *arXiv:1812.02428*.

[19] H. Chen, K. Laine, and R. Player, "Simple encrypted arithmetic library-seal v2. 1," in *Proc. Int. Conf. Financial Cryptography Data Security*, Springer-Verlag, 2017, pp. 3–18.

[20] M. R. Albrecht *et al.*, "Homomorphic encryption standard." *IACR Cryptol. ePrint Arch.*, vol. 2019, p. 939, 2019.

[21] S. Goldwasser and S. Micali, "Probabilistic encryption," *J. Comput. Syst. Sci.*, vol. 28, no. 2, pp. 270–299, 1984, doi: 10.1016/0022-0000(84)90070-9.

[22] C. R. Harris *et al.*, "Array programming with NumPy," *Nature*, vol. 585, no. 7825, pp. 357–362, Sep. 2020, doi: 10.1038/s41586-020-2649-2.

[23] M. S. Hossain and G. Muhammad, "Cloud-assisted speech and face recognition framework for health monitoring," *Mobile Netw. Appl.*, vol. 20, no. 3, pp. 391–399, 2015, doi: 10.1007/s11036-015-0586-3.

[24] Y. LeCun, "The MNIST database of handwritten digits," 1998. [Online]. Available: http://yann.lecun.com/exdb/mnist/

[25] H. Xiao, K. Rasul, and R. Vollgraf, "Fashion-MNIST: A novel image dataset for benchmarking machine learning algorithms," 2017.

[26] T. Morshed, M. M. Al Aziz, and N. Mohammed, "CPU and GPU accelerated fully homomorphic encryption," in *Proc. IEEE Int. Symp. Hardware Oriented Security Trust (HOST)*, 2020, pp. 142–153, doi: 10.1109/HOST45689.2020.9300288.

[27] M. O. Alberto Ibarrondo, Laurent Gomez, "Pyfhel: Python for homomorphic encryption libraries." 2018. [Online]. Available: https://github.com/ibarrond/Pyfhel

[28] A. Benaissa, B. Retiat, B. Cebere, and A. E. Belfedhal, "TenSEAL: A library for encrypted tensor operations using homomorphic encryption," 2021.

[29] A. Paszke *et al.*, "Pytorch: An imperative style, high-performance deep learning library," in *Proc. Adv. Neural Inf. Process. Syst.*, 2019, vol. 32, pp. 8026–8037.

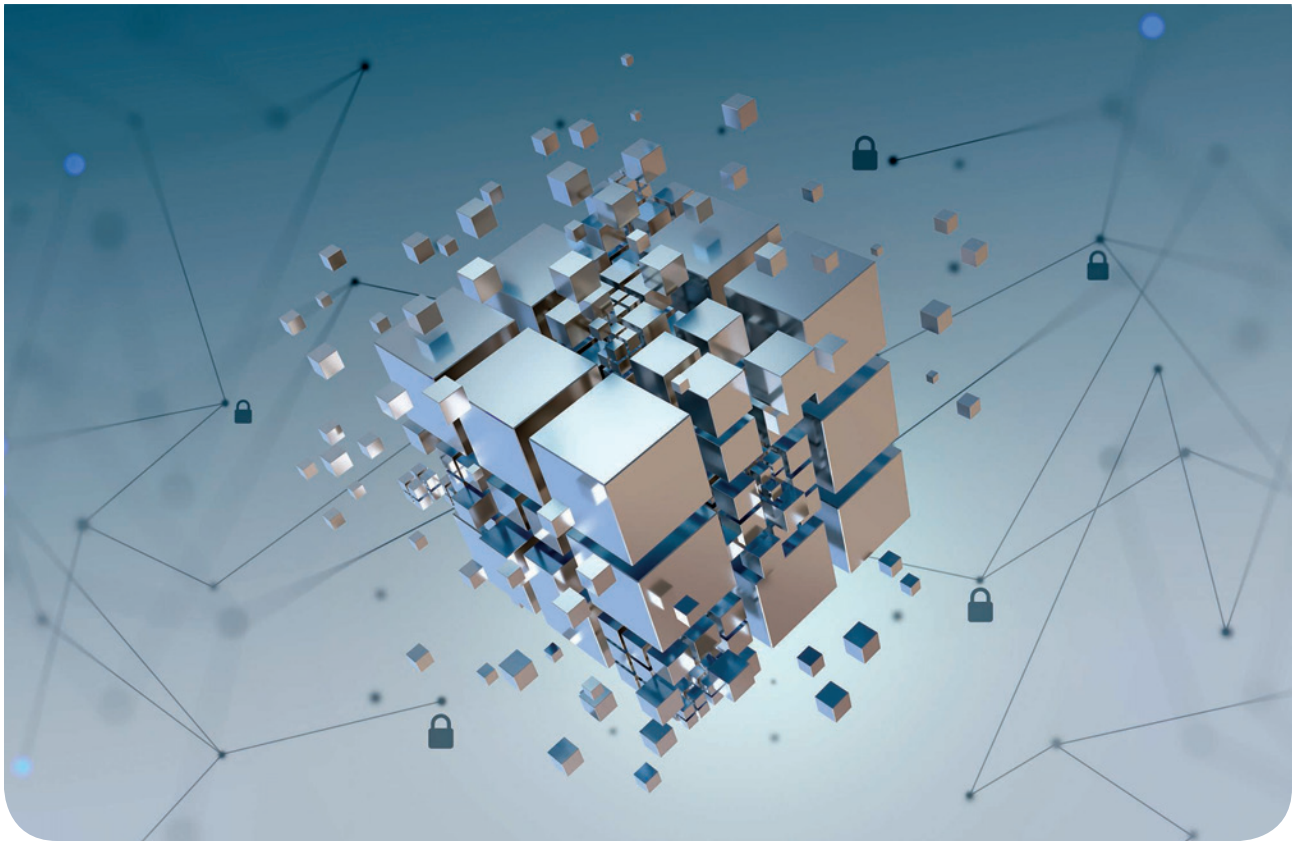
[30] S. Halevi and V. Shoup, "Algorithms in HElib," in *Proc. Annu. Cryptol. Conf.*, Springer-Verlag, 2014, pp. 554–571.

[31] I. Chillotti, M. Joye, D. Ligier, J.-B. Orfila, and S. Tap, "CONCRETE: Concrete operates on ciphertexts rapidly by extending TfhE," in *Proc. WAHC 2020–8th Workshop Encrypted Comput. Appl. Homomorphic Cryptography*, 2020, vol. 15.

[32] S. Disabato, A. Falcetta, A. Mongelluzzo, and M. Roveri, "A privacy-preserving distributed architecture for deep-learning-as-a-service," in *Proc. 2020 Int. Joint Conf. Neural Netw. (IJCNN)*, pp. 1–8, doi: 10.1109/IJCNN48605.2020.9207619.

[33] R. Dathathri *et al.*, "Chet: An optimizing compiler for fully-homomorphic neural-network inferencing," in *Proc. 40th ACM SIGPLAN Conf. Program. Lang. Des. Implementation*, 2019, pp. 142–156, doi: 10.1145/3314221.3314628.

When Federated Learning Meets Blockchain: A New Distributed Learning Paradigm



©SHUTTERSTOCK.COM/YURCHANKA SIARHEI

Chuan Ma, Jun Li, and Long Shi
Nanjing University of Science and Technology, CHINA

Ming Ding
CSIRO, AUSTRALIA

Taotao Wang
Shenzhen University, CHINA

Zhu Han
University of Houston, USA, and Kyung Hee University, SOUTH KOREA

H. Vincent Poor
Princeton University, USA

Digital Object Identifier 10.1109/MCI.2022.3180932
Date of current version: 19 July 2022

Abstract—Motivated by the increasingly powerful computing capabilities of end-user equipment, and by the growing privacy concerns over sharing sensitive raw data, a distributed machine learning paradigm known as federated learning (FL) has emerged. By training models locally at each client and aggregating learning models at a central server, FL has the capability to avoid sharing data directly, thereby reducing privacy leakage. However, the conventional FL framework relies heavily on a single central server, and it may fail if such a server behaves maliciously. To address this single point of failure, in this work, a blockchain-assisted decentralized FL framework is investigated, which can prevent malicious clients from poisoning the learning process, and thus provides a self-motivated and reliable learning environment for clients. In

Corresponding author: Jun Li (e-mail: jun.li@njust.edu.cn).

1556-603X/22©2022IEEE

this framework, the model aggregation process is fully decentralized and the tasks of training for FL and mining for blockchain are integrated into each participant. Privacy and resource-allocation issues are further investigated in the proposed framework, and a critical and unique issue inherent in the proposed framework is disclosed. In particular, a lazy client can simply duplicate models shared by other clients to reap benefits without contributing its resources to FL. To address these issues, analytical and experimental results are provided to shed light on possible solutions, i.e., adding noise to achieve local differential privacy and using pseudo-noise (PN) sequences as watermarks to detect lazy clients.

Future wireless networks are expected to require very low latencies and high reliability. Migrating machine learning (ML) to end-user equipments (UEs) promotes these requirements, giving them the capability of making decisions based on locally acquired data, even if it loses connectivity to the network. Since data available at a given end-user device is typically limited, the training of on-device ML models can benefit from data exchange among UEs [1].

However, directly exchanging data among UEs presents risks of privacy leakage and information hijacking [2]. To reduce this risk, federated learning (FL) has been proposed, which is an ML framework that trains an artificial intelligence (AI) model across multiple UEs holding local datasets. In particular, distributed UEs train ML models locally, sharing their model parameters with a central server where these local models are aggregated into a global model. In this way, FL allows UEs to cooperatively learn a global model without exchanging their data directly. FL has been applied in practical settings, including health care and autonomous driving [3].

Although FL offers advantages in latency and privacy enhancement, it suffers from several limitations. First, in the FL process, it is assumed that the aggregator is trustworthy and will make fair decisions in terms of user selection and aggregation. However, this assumption is not always satisfied in practical situations where a biased aggregator can intentionally favor a few selected UEs, thereby biasing learning performance [1]. Second, although the aggregator has access only to the models trained by its UEs, private client data can still be inferred from those models. Thus, if the aggregator is compromised, privacy leakage happens. Lastly, the conventional FL architecture is vulnerable to malicious clients that can attack learning via model poisoning [4].

As a secure technology, blockchain has the capability to tolerate a single point of failure with distributed consensus, and it can further implement incentive mechanisms to encourage participants to effectively contribute to the system [5]. For these reasons, blockchain has been introduced into FL to mitigate its aforementioned limitations. For example, [5] introduced a blockchained FL architecture to verify uploaded model parameters and investigated related system performance indices, such as learning delay and block generation rate. Moreover, [6] proposed a privacy-aware architecture that uses blockchain to enhance security when parameters of ML models are shared among UEs. In addition, the authors of [7] proposed a high-level framework by enabling encryption during model transmission, and [8] further applied this framework in a military setting. With the advanced features of blockchain, such as tamper-resistance, anonymity, and traceability, an immutable audit trail of ML models can be created for greater trustworthiness in tracking and proving provenance [9]. In addition, security and privacy issues arising in the decentralized FL framework are investigated in [6], [10], [11], which delegated the responsibility of storing ML models to a trust community in the blockchain. However, the assumption of a trust community may incur the same privacy issues when ML models are transmitted over the air, and the credibility of this community also needs further verification. In addition, these works either have not completely clarified and fully addressed incidental issues, such as the long learning delay and impact of blockchain forking on FL, or are difficult to apply.

In the present work, a blockchain-assisted decentralized FL (BLADE-FL) framework, which can overcome the single point of failure problem, is proposed in detail. In addition, several residual issues that exist in the BLADE-FL framework are further investigated, and related solutions are provided. The rest of this paper is organized as follows. The design of the BLADE-FL framework is presented in Sec. II, and residual issues, including privacy, resource allocation, and lazy clients, are investigated in Sec. III. In Sec. IV, extensive experimental results are provided to show the effectiveness of the corresponding solutions. Finally, promising future directions are suggested and conclusions are drawn in Sec.V.

II. BLADE-FL Framework

With the aid of blockchain, the aim is to build up a secure and reliable FL framework. To ensure this, the model updating process of FL is decentralized at each participating client, which is robust against the malfunction of traditional aggregators. In this section, the BLADE-FL framework, as well as how it achieves dynamic client selection and a decentralized learning aggregation process is presented.

The BLADE-FL framework is composed of three layers. In the network layer, the network features a decentralized peer-to-peer (P2P) network that consists of task publishers and training clients, wherein a learning mission is first published by a task publisher and then completed by the cooperation of several training clients. Different from previous work, in which model aggregation occurs in a trust community in the blockchain [5]–[11], a fully decentralized framework is realized in which each client must train ML models and mine blocks for publishing aggregating results. In the blockchain layer, each FL-related event, such as publishing a task, broadcasting learning models, and aggregating learning results, is tracked by blockchain. In the application layer, the smart contract (SC) and FL are utilized to execute the FL-related events. Next, the workflow and key components of the BLADE-FL framework are presented.

A. Workflow

As shown in Fig. 1, the workflow of the proposed framework consists of the following steps.

- ❏ **Step 1**: Task publishing and node selection. A task publisher broadcasts an FL task by deploying an SC over the blockchain network. In the deployed SC, the task publisher must deposit a reward as a financial incentive for the learning task. The SC selects available training nodes to participate in this learning task.
- ❏ **Step 2**: Local model broadcast. Each training client runs its local training by using its own data samples, and it broadcasts local updates and the corresponding processing information (e.g., computation time and local data size) over the P2P network. Privacy leakage may happen during this transmission, and this issue is further investigated in Sec. III-A.
- ❏ **Step 3**: Model aggregation. Upon receiving the local updates from other training nodes before a pre-set time-stamp, each client updates the global model according to the aggregating rule defined in the SC.
- ❏ **Step 4**: Block generation. Each training client changes roles from trainer to miner and begins mining until it either finds the required nonce or receives a generated block from other miners. The learning results are stored in the block as well. When one miner generates a new block, other clients verify the contents of this block (e.g., the nonce, state changed by the SC, transactions, and aggregated model). The resource-allocation issue happens in each client in this step, and related discussions will be given in Sec. III-B.
- ❏ **Step 5**: Block propagation. If a block is verified by the majority of clients, this block will be added on the blockchain and accepted by the entire network. The lazy client issue occurs in this step and is further investigated in Sec. III-C.
- ❏ **Step 6**: Global model download and update. Each training client downloads the aggregated model from the block and performs updates before the next round of learning.
- ❏ **Step 7**: Reward allocation. The SC deployed by the task publisher rewards the training clients according to their contributions to the learning task.

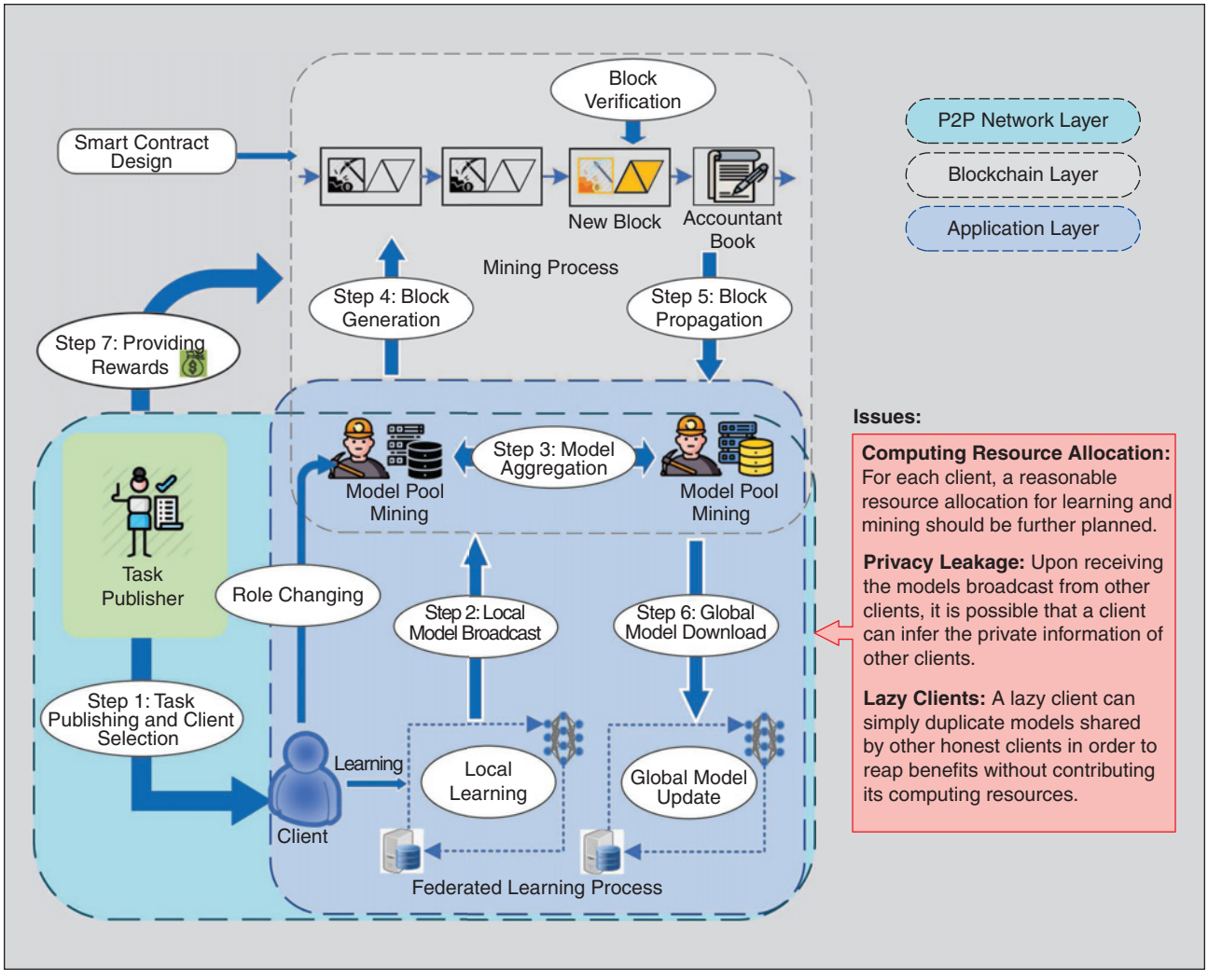


FIGURE 1 BLADE-FL workflow.

Before delving into each step, the key designs in BLADE-FL are elaborated as follows.

B. SC Design

SCs are self-executing contracts defining rules for negotiating, verifying the fulfillment of rules, and executing the agreement using formal code. The BLADE-FL framework relies on SCs to enable trusted dynamic client selections in terms of desired distributed learning services, without relying on a centralized authority. Moreover, BC-FL enables all clients to verify the learning results that are recorded on the blockchain, whereby distributed clients can be incentivized to participate and untrusted learning models can be detected. Based on the verification results, the reputation of each distributed client can be automatically updated, making the selection of learning nodes more reliable. In addition, the design of SCs in the BC-FL also includes the aggregating rules, and thus provides a fair and open rewarding of feedback for participating clients. The SC in BC-FL enables three main functions as follows.

Function 1: Learning task publishing. A task publisher broadcasts an FL task through an SC to all users. The SC contains the task requirements (e.g., the data size, training accuracy, latency, etc.), the aggregating rules, and rewards paid by the task publisher.

Function 2: Dynamic bidding for requests and automatic incentive. Distributed training nodes, acting as auctioneers, bid for the task by replying with their costs and capabilities. Note that to enforce accountability, each training client must stake a deposit to the SC. The task replies from training nodes are recorded on the blockchain by the SC. Then, the SC selects training clients with more valuable replies (e.g., higher capability and lower cost) as the bid winners to jointly execute the FL task. The training clients that lose the bidding will reclaim their deposits from the SC, while the deposits made by winners will be automatically refunded if the learning results are verified to be trustworthy afterwards.

Function 3: Learning results aggregation and rewards feedback. Before generating a new block, each client will aggregate the uploaded models according to the aggregating rule in the SC, in which the contribution of each one in the aggregated model is also recorded in the newly generated block. Then, the SC is automatically triggered to reward the miner that helps aggregate the learning model and the training clients that contribute to the FL process.

C. BLADE-FL Design

The main purpose of BLADE-FL is to enable trusted cooperative ML among distributed nodes. The decentralized accountability enables all miners to verify the quality of uploaded models that are recorded on the blockchain. In addition, distributed training nodes can be motivated to participate in the FL process and misbehaving ones can be recognized from the low-quality FL services they provide. The key steps follow.

As a secure technology, blockchain has the capability to tolerate a single point of failure with distributed consensus, and it can further implement incentive mechanisms to encourage participants to effectively contribute to the system [5].

Local model updating and uploading: Training nodes are bid winners with capable devices and available sets of data samples. In each learning iteration, each training node updates a local ML model in a parallel manner by using the global model and its local data samples, and broadcasts its local model in the network. In the present work, local updates can be received by all of the miners through the gossip protocol [12] over the P2P network. In this context, the aggregation process in traditional FL is decentralized to each client that stores the uploaded models in its respective model pool.

Model aggregation: After collecting the uploaded models in the pool, each client calculates the global model updates according to the aggregating rule in the SC. In the proposed architecture, the clients are designed to aggregate the learning parameters truthfully through a distributed ledger. Similar to the prevailing block structure in [6], each block in a ledger consists of body and header parts. Specifically, the body stores the local model updates, such as the local data size and computing time of the associated training node and the aggregated learning parameters. The header contains the information of a pointer to the previous block, block generation rate, and output value, such as the proof of work (PoW), in the consensus protocol.

Model recording and publishing: The clients record the aggregated models in their blocks and publish the recorded models by broadcasting the generated block to the entire network. The blocks can be generated by using distributed or lightweight consensus protocols, such as PoW, proof of stake (PoS), delegated PoS (DPoS), etc. [13]. In this paper, PoW is considered due to its strong security over decentralized networks, and a synchronous schedule is used to ensure that all of the miners start mining at the same time.

Once a client finds the hash value, its candidate block becomes a new block, and the generation rate of this block is controlled by the PoW difficulty. Then, this generated block is broadcast to all of the other miners in the framework. All of the other miners must verify the nonce and the aggregated results contained in this block. For example, clients can compare the aggregated results with the one in the publishing block or use a public testing dataset to justify the effectiveness of the uploaded models. If the verification result is correct, other clients will accept it as a legal block and record it; otherwise, others will discard this generated block and continue to mine the previous legal block.

Reward allocation: The task publisher provides learning rewards for the participating training nodes, and the

The decentralized accountability enables all miners to verify the quality of up-loaded models that are recorded on the blockchain.

volume can be proportional to the size of the training data. It is noted that the reward mechanism can be further amended by combining consideration of the data size and the quality of data samples. In this case, clients are responsible for verifying the trustworthiness of local updates after aggregation to address the situation that untruthful UEs may exaggerate their sample sizes with abnormal local model updates. Specifically, when clients calculate the rewards for each training node, they can give scores/reputations to the training nodes based on the model qualities. In the next aggregation, nodes with low scores will be given less weight, and they will be identified and gradually ignored during learning. In practice, this can be guaranteed by Intel's software guard extensions, allowing applications to be operated within a protected environment, which has already been used in blockchain technologies [14]. In addition, miners can also obtain rewards from mining and aggregating models, which can be treated as a gas tax in the traditional blockchain.

A task publisher first broadcasts an FL task through an SC to all of the clients. Consider that N clients dynamically bid for this task and use PoW as the consensus mechanism in the verification process. Thus, the pseudo-code of the proposed BLADE-FL framework is outlined in Algorithm 1.

ALGORITHM 1 BLADE-FL algorithm.

Data: Number of communication rounds T, initial model $\mathbf{w}^0$, and proximal term μ in local learning.

Task publishing and client selection.
Initialization: $t = 1$ and $\mathbf{w}_i^0 = \mathbf{w}^0$, $\forall i$
while $t \leq T$ **do**
  Local training:
  while $i \in \{1, 2, \ldots, N\}$ **do**
    Update local model $\mathbf{w}_i^{(t)}$ as
    $\mathbf{w}_i^t = \arg\min_{\mathbf{w}_i} \left(F_i(\mathbf{w}_i) + \frac{\mu}{2} || \mathbf{w}_i - \mathbf{w}^{t-1} ||^2 \right)$.
  Model broadcasting and aggregating:
  Update aggregated model $\mathbf{w}^{(t)}$ as
    $\mathbf{w}^t = \sum_{i=1}^{N} p_i \widetilde{\mathbf{w}}_i^t$.
  Block mining and verification:
  Each client starts mining its block that includes the aggregated model and verifies the generated block.
  Global model downloading:
  Each client downloads the aggregated model from the verified block and updates.
  Reward allocation for learning and mining.
  $t \leftarrow t + 1$
Result: $\widetilde{\mathbf{w}}^{(T)}$

III. Unique Issues and Potential Solutions

In this section, three critical issues that the proposed framework may be confronted with, namely, privacy, resource allocation, and lazy clients, are described.

A. Privacy

In BLADE-FL, the roles of each client include mining and training. To aggregate the global model, the trained local model will be published among clients, which raises privacy issues. Previous works [5]–[11] usually artificially assign the training and mining tasks to two disjoint sets of clients, and they widely adopt that the miners are always trustful. However, if an eavesdropper exists in the wireless environment, the published information of local models can cause privacy leakage. To address this, a differentially private mechanism can be implemented at the client side. In detail, the key steps are as follows.

- Each client sets up a self-required privacy level for itself before training. For example, the ith client may have a local privacy budget ϵ_i. Note that a small value of ϵ_i represents a high local privacy level, and it will induce more additive noises on the parameters.
- To achieve local differential privacy (LDP), each client will add a random noise that follows a certain distribution on the uploaded models. For example, a random Gaussian noise $N(0, \sigma^2)$ or a Laplace noise $Lap(\lambda)$ will be added. Note that a large noise power implies a high privacy level.
- Upon receiving the perturbed models, all of the clients can aggregate the global model locally and store it in the generated block. Because of the injected noise, the learning convergence as well as the system performance will be negatively affected. A trade-off between the privacy requirement and learning performance needs further investigation. In addition, a non-uniform allocation of additive noise over communication rounds may improve learning performance; for example, a decay rate for the noise power can be applied when the learning accuracy between two adjacent communication rounds stops improving [15].

B. Computing Resource Allocation

Since the computation resource is limited at each client, each participant must appropriately allocate the resources for local training and mining to complete the task. Specifically, more computing resources can be devoted either to accelerate model updating or block generation. To meet the specific task requirements, such as learning difficulty, accuracy, and delay, each node optimizes its allocation strategy to maximize its reward under constraints of local capability.

According to the constraints, the computing resource allocation can be formulated as an optimization problem under the accurate mathematical model, as follows, in detail.

- The block generation rate is determined by the computational complexity of the hash function and the total computing power of the blockchain network (i.e., total CPU cycles). The average CPU cycles required to generate a block can be defined as kc_B, where k denotes the mining difficulty and c_B is the average number of total CPU cycles required to generate a block. Thus, the average generation time of a block (t_B) can be expressed as kc_B/Nf, where N is the number of clients and f denotes the CPU cycles per second of each client.
- The training time consumed by each training iteration, t_T, can be expressed as $(|D|c_T/f)$, where $|D|$ denotes the number of samples of each client and c_T is the number of CPU cycles required to train one sample.
- Considering that a typical FL learning task is required to be accomplished within a fixed duration of T_{Sum}, the total learning and mining times should be satisfied that $K(\tau t_T + t_B) \leq T_{\text{Sum}}$, where K denotes the number of total communication rounds and τ the number of local training epochs. Thus, to achieve required learning performance, the number of communication rounds K should be optimized under a certain ratio between the training and mining time.

C. Lazy Clients

As the verification is processed locally, a lazy client may not perform local learning and directly copy uploaded parameters from other clients to save its computing resources. As a result, the client can devote more mining resources to reaping more mining rewards with a higher probability. However, this action degrades learning performance. To investigate the effect of lazy nodes on the system performance, related experimental results are provided in Sec.V-D.

To address the lazy client issue, a signature process can be implemented at each client, which is based on the pseudo-noise (PN) sequence. The signature in BLADE-FL is resilient to noise perturbation because the lazy clients are likely to perturb the plagiarized local models to hide their misbehavior. This process can improve detection accuracy of lazy clients at the cost of negligible overhead to the system. The details of the process are the following.

- Before broadcasting the local updates, each client will produce a PN sequence of length L, where L is usually a very large number (larger than the number of model parameters), and a same length with model parameters is selected and added to the local updates. This PN sequence has a high self-correlation coefficient and is difficult to detect or re-produce by other clients. Minimally, the complexity of detecting the PN sequence should be much larger than that of training the neural network so as to deter the attempt to discover the used PN sequence.
- Upon receiving local updates from the other clients, each client will use its own PN sequence to check the correlation coefficient with the updates. If high peaks in terms of the cross-correlation coefficient exist, then the lazy clients will be detected.
- Once a lazy client is recognized by a local client, this client can publish the previously used PN sequence to others and invite other honest clients to verify this process. Then, any future updates from the lazy client might be discarded as punishments.

To achieve local differential privacy, each client will add a random noise that follows a certain distribution on the uploaded models.

The pseudo-code of the PN sequence detection is shown in Algorithm 2.

IV. Experimental Results and Potential Solutions

In this section, several related experimental results are provided to show the issues in the multi-functional miner in the proposed BLADE-FL system.

A. System Setup

For each experiment, the original training data is divided into non-iid training sets, and locally computes a stochastic gradient descent (SGD) updated on each dataset, and then the server aggregates updates to train a globally shared classifier. The prototype is evaluated on the Fashion-MNIST and Cifar-10 datasets.

ALGORITHM 2 PN Sequence Detection Algorithm.

Data: Number of communication rounds T, number of clients N, detection threshold λ, and amplitude of PN sequence α.

Initialize: $\overline{\mathbf{w}}^0$, $t = 1$
while $t <= T$ **do**
  if $t = 1$ **then**
    i-th client adds an additive PN sequence to the learned model as $\widehat{\mathbf{w}}_i^t = \mathbf{w}_i^t + \alpha \mathbf{S}_i^t$.
    Calculates the cross-correlation value between model parameters and its own PN sequence as $\mathbf{C}_{ij}^t = \mathbf{S}_i^t * \widehat{\mathbf{w}}_j^t = \mathbf{S}_i^t * (\mathbf{w}_j^t + \alpha \mathbf{S}_j^t)$.
    Sums the cross-correlation value as $C_j^t = \sum_{i=1, i\neq j}^{N} C_{ij}^t$.
    Performs peak detection.
    if $C_j^t \geq \lambda$ **then**
      $\mathbf{w}_j^t$ is copied;
    else
      $\mathbf{w}_j^t$ is honest.
  Discards models that are detected as lazy clients
  Performs a global update as $\overline{\mathbf{w}}^t = \sum_{i=1}^{N^t} \left(|D_i| \widehat{\mathbf{w}}_i^t\right) / \left(\sum_{j=1}^{N^t} |D_j|\right)$, where N^t is number of honest clients after discarding lazy clients.
  $t = t + 1$
Result: $\overline{\mathbf{w}}^T$

In the following, the average results of 20 run experiments are collected. For the blockchain setup, the total computation resource is set to $T_{\text{Sum}} = 200$ for each training node, and the total number of clients to $N = 20$. In each communication round, each client uses t_B time resources to generate a block and t_T time resources to pursue a learning epoch, where $t_B = 2$ for all of the experiments. Letting $\theta = t_T/t_B$, a larger θ implies that the client allots more computing resources to learning in each communication round.

B. Investigation of Local Differential Privacy

Local differential privacy is applied to each client by adding random Gaussian noise to the uploaded models in each communication round. The testing accuracies of the Fashion-MNIST and Cifar-10 datasets are plotted in Fig. 2 with respect to different privacy levels ϵ. In addition, an adaptive noise decaying method is compared with the constant one, which will decrease the noise power when the accuracy stops increasing. The figure further shows that the system achieves higher performance with a larger value of ϵ, which is under weaker privacy protection, and the adaptive method can further improve the learning performance under the same level of privacy protection.

C. Investigation of Resource Allocation

In this subsection, the resource-allocation results and the training loss values with different ratios (θ) of both datasets are plotted in Fig. 3. The figure shows the system performance for different ratios as the number of total communication rounds increases. Usually, a lower loss function value represents better training performance. In detail, it can be found that there exists an optimal total communication round (K) for each computing ratio θ. For example, the smallest training-loss value can be obtained if clients stop learning in 14 communication rounds each with 15 learning epochs when $\tau = 1$ in the Fashion-MNIST dataset. Moreover, for different computing ratios, the optimal loss value tends to be different. This is due to the fact that the optimal number of local learning epochs varies according to different values of θ. In addition, similar trends can be found in the Cifar-10 dataset.

D. Investigation of Lazy Clients

In this subsection, the impact of lazy clients on the proposed framework is investigated. The signal-to-noise ratio (SNR) is used to denote the ratio of the power of original model parameters to

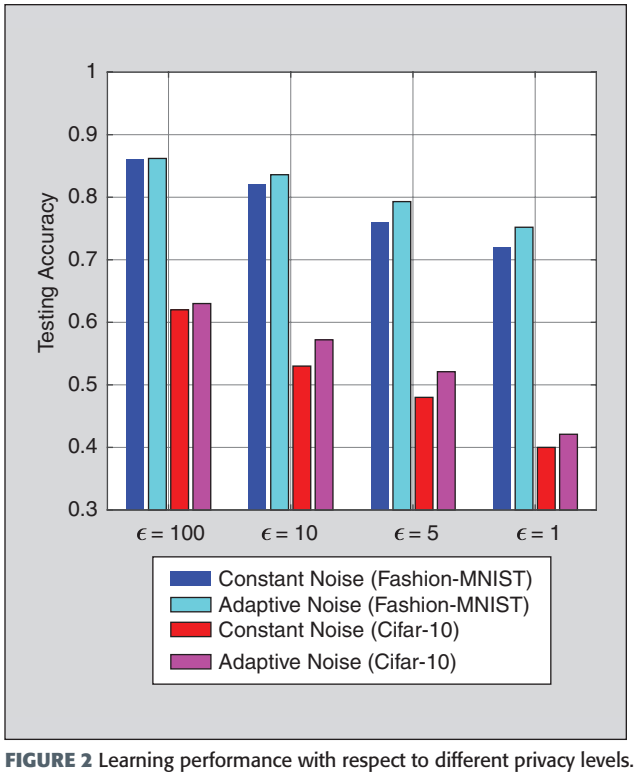


FIGURE 2 Learning performance with respect to different privacy levels.

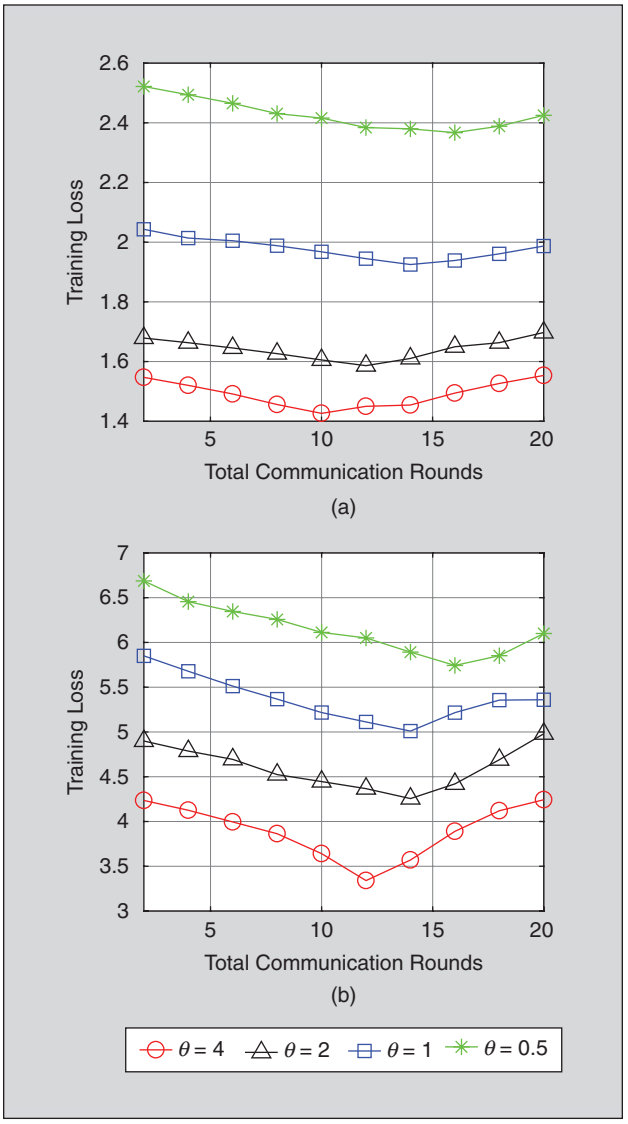


FIGURE 3 Learning performance of different total communication rounds under different resource-allocation ratios. (a) Fashion-Mnist (b) Cifar-10.

TABLE I Detection rate with different PN sequence powers for Fashion-MNIST and Cifar-10 datasets.

SNR	9 dB	6 dB	3 dB
Fashion-MNIST	0.931	0.989	0.999
Cifar-10	0.925	0.975	0.996

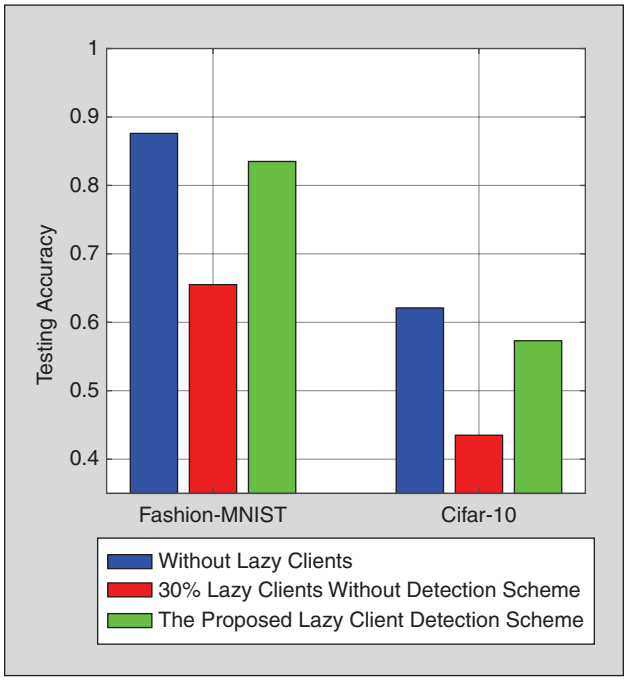


FIGURE 4 Learning performance with/without lazy client detection.

that of the injected PN sequence, and Table I represents the detection rate of lazy clients under different SNRs. If a high peaks in terms of the cross-correlation coefficient surpasses a predefined threshold, this client is identified as a lazy one. A PN sequence of length 2^{15} is generated and the first 25400 values are used to add onto the parameters. From the results with different SNRs, the detection performance is remarkable, and a nearly 100% rate of lazy client recognition when SNR=3 dB can be obtained. Fig. 4 shows the PN sequence-protection performance (SNR = 6 dB) when there are 30% (6) lazy clients in each communication round. As can be seen in this figure, the system performance with a certain percentage of lazy clients degrades sharply, i.e., 22.1% and 19.6% reduction for the Fashion-MNIST and Cifar-10 datasets, respectively. In addition, the proposed PN sequence-protection method achieves 18% and 13.8% performance gain for each dataset, respectively.

V. Future Directions and Conclusions

In this paper, the weaknesses of FL have been reviewed and a blockchain-assisted decentralized FL architecture, called BLADE-FL, has been proposed to address some of these weaknesses. The effectiveness of BLADE-FL has been shown in addressing these issues, notably the problem of a single point of failure that exists in conventional FL. In addition, further issues have been investigated for BLADE-FL including privacy, resource allocation, and lazy clients, and possible solutions have been provided to address those issues and explored with experiments. These results provide guidelines for the design of the BLADE-FL framework.

Some directions for further study in this area include asynchronous and heterogenetic investigations for different client capabilities, such as computing capability, training-data size, and transmitting diversity, and SC design, which provides reasonable reward allocation for training and mining. In addition, lightweight model transmission using quantization and sketch may be an alterative way of reducing the transmission cost.

Acknowledgments

This work was supported in part by the National Natural Science Foundation of China under Grant No. 62002170, and 61872184, and in part by the Fundamental Research Funds for the Central Universities under Grant No. 30919011274, in part by the Natural Science Foundation of Jiangsu Province under Grant BK20210331, in part by the Jiangsu Specially-Appointed Professor Program in 2021, in part by the Natural Science Fund of Guangdong Province under Grant 2020A1515010708 and the Natural Science Fund of Shenzhen under Grant JCYJ20210324094609027, and in part by the U.S. National Science Foundation under Grants ECCS-2039716, CNS-2107216 and CNS-2128368.

References

[1] C. Ma *et al.*, "On safeguarding privacy and security in the framework of federated learning," *IEEE Netw.*, vol. 34, no. 4, pp. 242–248, Jul./Aug. 2020, doi: 10.1109/MNET.001.1900506.
[2] Z. Liu, P. Longa, G. C. C. F. Pereira, O. Reparaz, and H. Seo, "Four$\mathbb{Q}$ on embedded devices with strong countermeasures against side-channel attacks," *IEEE Trans. Dependable Secure Comput.*, vol. 17, no. 3, pp. 536–549, May/Jun. 2020, doi: 10.1007/978-3-319-66787-4_32.
[3] Y. Chen, X. Qin, J. Wang, C. Yu, and W. Gao, "Fedhealth: A federated transfer learning framework for wearable healthcare," *IEEE Intell. Syst.*, vol. 35, no. 4, pp. 83–93, Jul./Aug. 2020, doi: 10.1109/MIS.2020.2988604.
[4] T. Li, A. K. Sahu, A. Talwalkar, and V. Smith, "Federated learning: Challenges, methods, and future directions," *IEEE Signal Process. Mag.*, vol. 37, no. 3, pp. 50–60, May 2020, doi: 10.1109/MSP.2020.2975749.
[5] H. Kim, J. Park, M. Bennis, and S. Kim, "Blockchained on-device federated learning," *IEEE Commun. Lett.*, vol. 24, no. 6, pp. 1279–1283, Jun. 2020, doi: 10.1109/LCOMM.2019.2921755.
[6] Y. Lu, X. Huang, Y. Dai, S. Maharjan, and Y. Zhang, "Blockchain and federated learning for privacy-preserved data sharing in industrial IoT," *IEEE Trans. Ind. Informat.*, vol. 16, no. 6, pp. 4177–4186, Jun. 2020, doi: 10.1109/TII.2019.2942190.
[7] X. Bao, C. Su, Y. Xiong, W. Huang, and Y. Hu, "Flchain: A blockchain for auditable federated learning with trust and incentive," in *Proc. 2019 5th Int. Conf. Big Data Comput. Commun. (BIGCOM)*, pp. 151–159, doi: 10.1109/BIGCOM.2019.00030.
[8] P. K. Sharma, J. H. Park, and K. Cho, "Blockchain and federated learning-based distributed computing defence framework for sustainable society," *Sustain. Cities Soc.*, vol. 59, p. 102,220, 2020.
[9] S. Wang, "Blockfedml: Blockchained federated machine learning systems," in *Proc. 2019 Int. Conf. Intell. Comput., Autom. Syst. (ICICAS)*, pp. 751–756, doi: 10.1109/ICICAS48597.2019.00162.
[10] Y. Qu, S. R. Pokhrel, S. Garg, L. Gao, and Y. Xiang, "A blockchained federated learning framework for cognitive computing in industry 4.0 networks," *IEEE Trans. Ind. Informat.*, vol. 17, no. 4, pp. 2964–2973, Apr. 2021, doi: 10.1109/TII.2020.3007817.
[11] S. Otoum, I. Al Ridhawi, and H. Mouftah, "Blockchain-supported federated learning for trustworthy vehicular networks," Dec. 2020, pp. 1–6.
[12] M. Jelasity, "Gossip," in *Self-Organising Software*. New York, NY, USA: Springer-Verlag, 2011, pp. 139–162.
[13] L. Ismail, H. Materwala, and S. Zeadally, "Lightweight blockchain for healthcare," *IEEE Access*, vol. 7, pp. 149,935–149,951, 2019, doi: 10.1109/ACCESS.2019.2947613.
[14] P. Fairley, "Blockchain world - feeding the blockchain beast if bitcoin ever does go mainstream, the electricity needed to sustain it will be enormous," *IEEE Spectr.*, vol. 54, no. 10, pp. 36–59, Oct. 2017, doi: 10.1109/MSPEC.2017.8048837.
[15] K. Wei *et al.*, "User-level privacy-preserving federated learning: Analysis and performance optimization," *IEEE Trans. Mobile Comput.*, early access, Feb. 4, 2021, doi: 10.1109/TMC.2021.3056991.

Yansen Su, Zhongxiang Jin, Ye Tian, and Xingyi Zhang
Anhui University, CHINA

Kay Chen Tan
The Hong Kong Polytechnic University, HONG KONG SAR

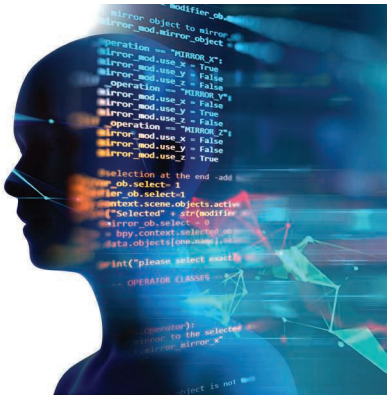
Comparing the Performance of Evolutionary Algorithms for Sparse Multi-Objective Optimization via a Comprehensive Indicator

Abstract

Many real-world multi-objective optimization problems (MOPs) are characterized by a large number of decision variables, where the decision variables are mostly set to zero in the Pareto optimal solutions. Although some multi-objective evolutionary algorithms (MOEAs) have been tailored for large-scale MOPs in recent years, most of them do not consider the sparse nature of Pareto optimal solutions, and their effectiveness to sparse MOPs has not been investigated. Therefore, this work aims to compare the performance of MOEAs on sparse MOPs by suggesting a comprehensive performance indicator. In comparison to existing indicators assessing the convergence and diversity of a solution set according to predefined reference points, the proposed indicator can assess the convergence, diversity, and sparsity without using any reference point. Based on the proposed indicator, an experiment is conducted to compare the performance of 11 state-of-the-art MOEAs on 60 test instances taken from benchmark suites and real-world applications. The statistical results show that some MOEAs are significantly better than the others for solving sparse MOPs, and the proposed indicator is effective for the performance assessment on sparse MOPs.

I. Introduction

Multi-objective optimization problems (MOPs) widely exist in scientific and engineering areas, which contain two or more objectives to be optimized simultaneously [1]. In the absence of a single solution making all the objectives optimal, there exist multiple Pareto optimal solutions for an MOP, which make diverse trade-offs between the conflicting objectives. Since multiple optimal solutions rather than a single one need to be provided, solving MOPs is much more difficult than solving single-objective optimization problems. More seriously, with the curse of dimensionality in large-scale optimization, it is extremely difficult for conventional MOEAs to find multiple optimal solutions for large-scale MOPs [2].



©SHUTTERSTOCK.COM/WHITEMOCCA

To improve the efficiency of approximating Pareto optimal solutions from high-dimensional decision spaces, a number of MOEAs have been tailored for solving large-scale MOPs since 2013 [2]. By adopting the divide-and-conquer strategy for large-scale single-objective optimization [3], some early MOEAs for large-scale MOPs divide the decision variables randomly or heuristically and optimize each group of variables alternately [2], [4]. While the variable grouping strategies for single-objective optimization do not preserve the population diversity in the objective space, some MOEAs group the decision variables according to their contributions to the convergence and diversity of the population, showing high performance on large-scale MOPs with complicated landscapes [5], [6]. On the other hand, some MOEAs aim to facilitate the solving of large-scale MOPs by converting the original problem into a small-scale one, where quasi-optimal solutions can be obtained with a few function evaluations [7], [8]. Similarly, some MOEAs adopt the dimensionality reduction strategies used in machine learning to directly reduce the decision space for an improvement of the convergence speed [9], [10]. In addition to the variable grouping and dimensionality reduction based MOEAs, some others suggest novel variation operators [11], [12] or probability models [13], [14] to search for optimal solutions directly, which are versatile for solving large-scale MOPs with different variable linkages.

In spite of the promising performance of existing large-scale MOEAs on a variety of benchmark problems, only few of them have been employed to tackle the large-scale MOPs in real-world applications. In addition to the high budget of evaluations required by these MOEAs, they do not take the characteristics of real-world applications into account, such as the sparse nature of Pareto optimal solutions. For the large-scale MOPs in many fields including machine learning [15], data mining [16],

Digital Object Identifier 10.1109/MCI.2022.3180913
Date of current version: 19 July 2022

Corresponding author: Ye Tian (e-mail: field910921@gmail.com).

1556-603X/22©2022IEEE

network science [17], signal processing [18], and economics [19], their Pareto optimal solutions are generally very sparse in the decision space, i.e., most decision variables of the Pareto optimal solutions are zero. Since it is difficult for existing large-scale MOEAs to generate many decision variables of zeros, several MOEAs have been tailored for solving large-scale sparse MOPs in recent years [20]–[22]. By suggesting novel variation operators and dimensionality reduction strategies, these MOEAs can not only highly reduce the decision space of large-scale MOPs, but also maintain the sparsity of solutions.

In comparison to other large-scale MOEAs, the above MOEAs can obtain sparser and better-converged solutions for large-scale sparse MOPs. However, a deep comparative analysis of them and other large-scale MOEAs has not been performed, and the performance of state-of-the-art MOEAs on popular real-world applications is still unclear. More seriously, most existing performance indicators are unsuitable for assessing the performance of MOEAs on sparse MOPs. On the one hand, a set of reference points on the true Pareto front are required by some indicators (e.g., IGD [23]), while the true Pareto front is unknown for real-world MOPs; although some other indicators (e.g., HV [24]) require a single reference point not on the true Pareto front, the reference point is still difficult to be set for fairness [25]. On the other hand, existing performance indicators assess only the convergence and diversity of solutions in the objective space, without considering the sparsity of solutions in the decision space for a more accurate performance assessment on sparse MOPs.

To address the above issues, this work proposes a comprehensive performance indicator for studying the performance of state-of-the-art MOEAs on large-scale sparse MOPs. The proposed indicator can assess the convergence, diversity, and sparsity of multiple solution sets simultaneously, which first sorts the solutions sets into several levels according to their dominance relations and sparsity, then tunes the rankings of the solution sets according to their diversity. More importantly, the proposed indicator does not require the assistance of any reference point or parameter, which facilitates the performance comparisons on real-world applications. In the experiments, we study the performance of two classical MOEAs, six large-scale MOEAs, and three sparse MOEAs on 60 test instances taken from eight benchmark large-scale sparse MOPs and seven real-world applications. The experimental results indicate that the large-scale MOEAs do not outperform classical MOEAs, while the sparse MOEAs are obviously better than classical MOEAs and large-scale MOEAs for solving large-scale sparse MOPs. Moreover, the experiments also verify that the proposed indicator is more effective than existing indicators in assessing the solution sets for sparse MOPs.

The rest of this paper is organized as follows. Section II introduces the large-scale sparse MOPs in practical applications, and reviews the state-of-the-art MOEAs for solving large-scale sparse MOPs and popular performance indicators for multi-objective optimization. Section III is devoted to the description of the proposed indicator. Section IV presents the experimental results and gives some analysis. Finally, conclusions are drawn in Section V.

II. Related Work

A. Large-Scale Sparse Multi-Objective Optimization Problems

An unconstrained MOP can be mathematically defined as:

$$\begin{aligned} &\min \quad \mathbf{f}(\mathbf{x}) = (f_1(\mathbf{x}), f_2(\mathbf{x}), \ldots, f_M(\mathbf{x})) \\ &\text{s. t.} \quad \mathbf{x} = (x_1, x_2, \ldots, x_D) \in \Omega, \end{aligned} \tag{1}$$

where solution $\mathbf{x}$ denotes a decision vector containing D variables, $\mathbf{f}(\mathbf{x})$ denotes its objective vector containing M conflicting objectives, and Ω is the decision space. A solution $\mathbf{x}$ is said to dominate and be better than another solution $\mathbf{y}$ (denoted as $\mathbf{x} \prec \mathbf{y}$), if and only if

$$\begin{cases} f_i(\mathbf{x}) \le f_i(\mathbf{y}), & \forall i = 1, 2, \ldots, M \\ f_j(\mathbf{x}) < f_j(\mathbf{y}), & \exists j = 1, 2, \ldots, M \end{cases}. \tag{2}$$

The solutions that are not dominated by any other solutions in Ω are the Pareto optimal solutions for the MOP, and the objective vectors of all the Pareto optimal solutions constitute the Pareto front of the MOP. Since the number of Pareto optimal solutions may be infinite for continuous MOPs, the goal of solving an MOP is to obtain a solution set A as an approximation of all the Pareto optimal solutions, where the distance between A and the Pareto front is denoted as its convergence, and the spread and evenness of A are denoted as its diversity [26]. For multi-objective optimization, both the convergence and diversity of A should be as good as possible.

An MOP with a large number of decision variables (i.e., $D \ge 100$ in general) is called a large-scale MOP, and an MOP whose Pareto optimal solutions contain many variables of zeros is known as a sparse MOP. For the MOPs in many scientific and engineering fields, they are characterized by both a large number of decision variables and sparse Pareto optimal solutions. Firstly, the objective functions of many real-world MOPs are calculated based on large datasets (e.g., training sets in neural network training [15], transaction datasets in pattern mining [16], graphs in critical node detection [17], and expected returns in portfolio optimization [19]), which introduce many decision variables to be optimized (e.g., weights in neural network training, items in frequent pattern mining, nodes in critical node detection, and ratios of instruments in portfolio optimization). Secondly, most decision variables of the Pareto optimal solutions are zero, which is determined by the objective functions of many real-world MOPs. For some MOPs, the sparsity of solutions is regarded as an objective to be optimized (e.g., network complexity in neural network training and the number of selected nodes in critical node detection), which leads to lightweight solutions to be efficiently implemented in practice. Although some other MOPs do not optimize the sparsity of solutions explicitly, the Pareto optimal solutions are still very sparse due to the restriction of objectives, since these MOPs are subset

selection problems and selecting only a small proportion of elements can lead to good objective values (e.g., support in pattern mining and total return in portfolio optimization). As a consequence, large-scale sparse MOPs widely exist in the real world, where a list of some popular applications is presented in [22].

The difficulties of large-scale sparse MOPs mainly lie in the high-dimensional decision spaces and the relatively expensive function evaluations. Owing to the stochastic search strategies of MOEAs, they will suffer from the curse of dimensionality when solving large-scale MOPs, where much more function evaluations are required to search in a higher-dimensional decision space [27]. Considering the expensiveness of the function evaluations, it is unaffordable and impractical for conventional MOEAs to solve large-scale sparse MOPs properly, and novel search strategies should be customized to save the function evaluations.

B. MOEAs for Large-Scale MOPs and Sparse MOPs

The search strategies in existing large-scale MOEAs are mainly based on three ideas, i.e., variable grouping, dimensionality reduction, and novel variation operators. The variable grouping based MOEAs aim to divide the decision variables into multiple groups and optimize each group alternately, so that the high-dimensional decision space is converted into several low-dimensional spaces. These MOEAs suggest different grouping strategies to strike a balance between efficiency, convergence, and diversity, such as the random grouping in CCGDE3 [2], the control variable analysis in MOEA/DVA [5], the differential grouping in CCLSM [28], and the variable clustering in LMEA [6]. The dimensionality reduction based MOEAs directly reduce the dimensionality of the decision space, thus quickly navigating to optimal subspaces and accelerating the convergence speed. These MOEAs learn optimal subspaces from the current population via different strategies, such as the problem transformation in WOF [7], the random embedding in ReMO [9], the problem reformulation in LSMOF [8], and the principal component analysis in PCA-MOEA [10]. The novel variation operator based MOEAs enhance the search ability of conventional variation operators (e.g., those in genetic algorithm [29], differential evolution [30], and particle swarm optimization [31]) by proposing new operators, such as the Gaussian process based inverse model in IM-MOEA [32], the competitive swarm optimizer in LMOCSO [11], the adaptive offspring generation in DGEA [12], and the covariance matrix adaptation evolution strategy in S^3-CMA-ES [14].

Although these large-scale MOEAs can be employed to solve large-scale sparse MOPs in theory, it is difficult for them to directly find the exact optimal values of most decision variables (i.e., zero) due to the stochastic search paradigm. On the contrary, an algorithm is required to find the decision variables that should be zero and optimize the other decision variables. For this aim, SparseEA [20] suggests a bi-level encoding scheme to represent the solutions for sparse MOPs, which includes a binary vector $\mathbf{xb} = (xb_1, xb_2, \ldots)$ denoting whether each decision variable should be zero and a real vector $\mathbf{xr} = (xr_1, xr_2, \ldots)$ denoting the value of each decision variable, where the decision variables x_1, x_2, for function evaluations are obtained by

$$x_i = xb_i \times xr_i, \ i = 1, 2, \ldots, D. \tag{3}$$

By optimizing the real vector and binary vector with different genetic operators, SparseEA can search for the zero decision variables and optimize the other decision variables simultaneously. Moreover, SparseEA generates the binary vectors through a new population initialization strategy, a crossover operator, and a mutation operator, which can maintain the sparsity of solutions. To further improve the efficiency, MOEA/PSL [22] adopts two unsupervised neural networks (i.e., restricted Boltzmann machine [33] and denoising autoencoder [34]) to reduce the dimensionality of the binary vector and the real vector, respectively. Hence, a sparse distribution and a compact representation of the decision variables can be learnt from the current population. Besides, MOEA/PSL is equipped with a parameter adaptation strategy for automatically determining the parameters in training the neural networks. Similarly, PM-MOEA [27] suggests an evolutionary pattern mining approach to reduce the dimensionality of the binary vector, which is parameterless and provides better diversity than neural networks. PM-MOEA also proposes an unbalanced crossover operator and an unbalanced mutation operator, using different probabilities to flip the binary variables in **xb** to ensure the sparsity of offspring solutions. In addition to the above three MOEAs, some work also considers the sparsity of solutions in multimodal optimization [35] and expensive optimization [21] in recent years.

To sum up, large-scale MOEAs converge faster than conventional MOEAs due to the grouping of variables, reduction of decision space, and novel variation operators. However, for sparse MOPs with real variables (e.g., neural network training and portfolio optimization), large-scale MOEAs can hardly find the optimal values of zeros. For sparse MOPs with binary variables (e.g., patterning mining and critical node detection), they are not applicable since the search strategies tailored for large-scale optimization can only work in continuous spaces. By contrast, sparse MOEAs can easily generate sparse solutions with real or binary variables, but they are ineffective for solving the MOPs without sparse Pareto optimal solutions. The applicability of some representative MOEAs to large-scale sparse MOPs is summarized in Table I.

C. Performance Indicators for Multi-Objective Optimization

To investigate the performance of MOEAs, the quality of the solution sets obtained by MOEAs should be quantified by tailored performance indicators [36], which can be divided into three

categories. The first category of indicators assesses the convergence of a solution set A, for instance, RNI (i.e., ratio of non-dominated individuals) [37] calculates the ratio of non-dominated solutions in A, Purity [38] counts the solutions in A that are non-dominated with those in other solution sets, and GD (i.e., generational distance) [39] measures the mean distance between each solution in A to the reference points on the true Pareto front. The second category assesses the diversity of a solution set A, for example, Spacing [40] calculates the standard deviation of the minimum distances of each solution to the others in A, CL_μ [41] counts the number of hypercubes having at least one solution, and CPF (i.e., coverage over Pareto front) [26] calculates the coverage of A over the reference points on the true Pareto front. The third category assesses both the convergence and diversity of a solution set, such as IGD (i.e., inverted generational distance) [23] calculating the mean distance between each reference point on the true Pareto front and the solutions in A, and HV (i.e., hypervolume) [24] calculating the area covered by A with respect to a reference point.

The third category of indicators is the most widely used since both the convergence and diversity should be considered in multi-objective optimization. However, IGD requires a set of uniformly distributed reference points on the true Pareto front, which is difficult to be sampled for MOPs with irregular Pareto front and impossible for real-world MOPs whose Pareto fronts are unknown [42]. HV requires a reference point consisting of slightly larger objective values than the solutions, which is also difficult to be determined for a relatively fair comparison, where the variation of the reference point may change the performance rankings of multiple MOEAs [25]. Therefore, it is not easy to use these indicators in practice, especially for engineers not familiar with MOEAs. Moreover, most indicators assess the convergence and diversity of solutions in the objective space, while only few consider the decision variables of solutions [43], and none of them consider the sparsity of solutions. Since sparse solutions contain a few elements corresponding to easy implementation and high efficiency, they are likely to be preferred by decision makers. Therefore, the sparsity of solutions should also be considered as a criterion when assessing the performance of MOEAs on sparse MOPs.

In view of the limitations of existing indicators, this work proposes a new performance indicator for the performance assessment on sparse MOPs in Section III, where the convergence, diversity, and sparsity of multiple solution sets can be assessed without using any reference point. Then, the proposed indicator is used to assess the performance of 11 MOEAs on 60 large-scale sparse MOPs in Section IV.

III. The Proposed Performance Indicator

A. Main Idea of the Proposed Indicator

Unlike the convergence and diversity that can be simultaneously assessed in the objective space (e.g., by using IGD and HV), the sparsity of solutions is solely assessed in the decision space. Hence, it has to separately assess the convergence, diversity, and sparsity by different criteria and integrate them into a scalar. Considering that the primary goal is to minimize the objectives and the sparsity is a pivotal factor in decision making, the convergence and sparsity are given priority over the diversity. Besides, it is unreasonable to assign different weights to the three criteria and sum

TABLE I Applicability of some representative MOEAs to large-scale sparse MOPs.

		SUITABLE FOR		
ALGORITHM	**MAIN SEARCH STRATEGY**	**LARGE-SCALE MOPs**	**BINARY MOPs**	**SPARSE MOPs**
NSGA-II [29]	Crossover and mutation		√	
MOEA/D-DE [30]	Differential evolution			
CCGDE3 [2]	Random grouping	√		
CCLSM [28]	Differential grouping	√		
MOEA/DVA [5]	Variable analysis	√		
LMEA [6]	Variable clustering	√		
WOF [7]	Problem transformation	√		
ReMO [9]	Random embedding	√		
PCA-MOEA [10]	Principal component analysis	√		
LSMOF [8]	Problem reformulation	√		
IM-MOEA [32]	Gaussian process	√		
LMOCSO [11]	Competitive swarm optimizer	√		
DGEA [12]	Adaptive offspring generation	√		
S^3-CMA-ES [14]	Covariance matrix adaptation evolution strategy	√		
SparseEA [20]	Bi-level encoding	√	√	√
MOEA/PSL [22]	Unsupervised neural networks	√	√	√
PM-MOEA [27]	Pattern mining	√	√	√

them up, as they have different magnitudes and the predefined weights will introduce bias.

On the other hand, the reference points are generally inevitable in the assessment of convergence and diversity. Some indicators (e.g., GD and IGD) measure the difference between a solution set and a set of reference points, where the reference points should be uniformly sampled on the Pareto front. Some others (e.g., CL_μ and HV) calculate the area covered by a solution set with respect to a reference point, which can be the nadir point (i.e., the point consisting of the maximum objective values of the Pareto front) of the problem. While the reference points required by these unary indicators are difficult to be determined in practice, it is desirable to design a polynary indicator to compare multiple solution sets with each other. That is, the reference points for each solution set consist of all the other compared solution sets, and any other reference point is no longer required.

To meet the above considerations, this work proposes a performance indicator to assess the convergence, sparsity, and diversity of the solution sets for sparse MOPs, termed CSD. Given multiple solution sets, the proposed CSD first assigns them into several levels according to their convergence and sparsity, then sorts the solution sets in the same level according to their diversity. This way, the convergence and sparsity are prioritized by the proposed indicator and all the three criteria can be easily integrated into a scalar. Moreover, no additional weight or reference point is required.

B. Criteria for Assessing Convergence, Sparsity, and Diversity

The Pareto dominance introduces partial orders between solutions, and solutions can be sorted into several levels according to their dominance relations (i.e., using non-dominated sorting [44]). Similarly, the proposed CSD sorts multiple solution sets according to the dominance relations between the solutions in each two sets. More specifically, a non-dominated solution set A is superior over a non-dominated solution set B in terms of convergence if

$$\begin{cases} |\{\mathbf{y}\in B \mid \exists \mathbf{x}\in A : \mathbf{x}\prec \mathbf{y}\}| > 0.5\,|B| \\ |\{\mathbf{x}\in A \mid \exists \mathbf{y}\in B : \mathbf{x}\prec \mathbf{y}\}| > 0.5\,|A| \end{cases}. \tag{4}$$

That is, A is superior over B if more than half the solutions in B are dominated by at least half the solutions in A. Obviously, the above relation is irreflexive, antisymmetric, and nontransitive.

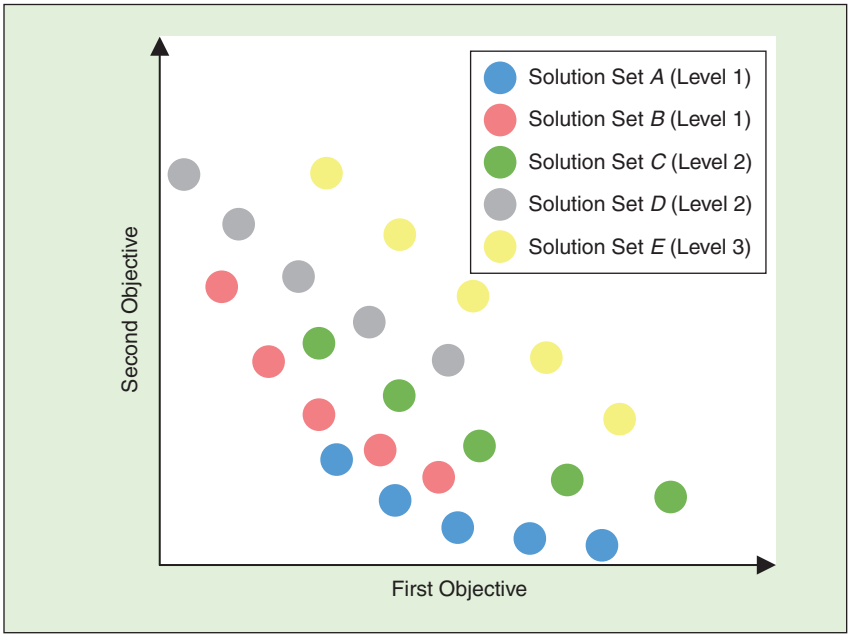


FIGURE 1 Levels of five non-dominated solution sets in terms of convergence.

The theoretical proof of the antisymmetric nature is given as follows.

Proof: Assuming that two non-dominated solution sets P and Q are superior over each other, according to the second condition in (4) when P is superior over Q:

$$|S_1| = |\{\mathbf{x}\in P \mid \exists \mathbf{y}\in Q : \mathbf{x}\prec \mathbf{y}\}| > 0.5\,|P|. \tag{5}$$

Besides, according to the first condition in (4) when Q is superior over P:

$$|S_2| = |\{\mathbf{x}\in P \mid \exists \mathbf{y}\in Q : \mathbf{y}\prec \mathbf{x}\}| > 0.5\,|P|. \tag{6}$$

Since $|S_1| > 0.5\,|P|, |S_2| > 0.5\,|P|$, and $S_1, S_2 \subseteq P$, we have

$$S_1 \cap S_2 \neq \emptyset. \tag{7}$$

Let $\mathbf{x}\in S_1 \cap S_2$, according to the definition of S_1, there exists a $\mathbf{y}\in Q$ satisfying $\mathbf{x}\prec \mathbf{y}$. Besides, according to the definition of S_2, there exists a $\mathbf{y}'\in Q$ satisfying $\mathbf{y}'\prec \mathbf{x}$. That is,

$$\mathbf{y}'\prec \mathbf{x}\prec \mathbf{y}, \tag{8}$$

which is contradictory with the precondition that B is a non-dominated solution set. Therefore, two non-dominated solution sets cannot be superior over each other. ■

In [45], solution set A outperforms solution set B only if all the solutions in B are dominated by those in A. While such a definition is so strict that all the solution sets may not be superior over any other, the ratio of dominated solutions in B should be set to a small value. On the other hand, the relation may be symmetric (i.e., A and B are superior over each other) if the ratio of dominated solutions is smaller than 0.5. Hence, the ratio is set to 0.5 here for the sake of distinguishability and antisymmetry.

After determining the relations between each two solution sets, the solution sets not inferior to any others are assigned to the first level and temporarily ignored. Then, the remaining solution sets not inferior to any others are assigned to the second level, and the operation repeats until all the solution sets are assigned. As illustrated in Fig. 1, solution set A is assigned to the first

level since all its solutions are nondominated, and solution set B is also assigned to the first level since fewer than half the solutions in B are dominated by those in A. In fact, all the nondominated solutions in A and B constitute the Pareto front and it is reasonable to assign both solution sets to the first level. By contrast, solution set C contains four solutions dominated by those in A and solution set D contains four solutions dominated by those in B, hence both C and D are assigned to the second level. In addition, solution set E has four solutions dominated by those in C and is assigned to the third level.

In terms of sparsity, it originally indicates the ratio of zero variables in a decision vector. Thus, the sparsity of a solution set A can be defined as

$$sparsity(A) = 1 - \frac{1}{|A|}\frac{1}{D}\sum_{\mathbf{x}\in A}\|\mathbf{x}\|_0, \quad (9)$$

where D denotes the number of variables and $\|\mathbf{x}\|_0$ denotes the number of nonzero variables in $\mathbf{x}$. While the sparsity of solution sets is continuous, it should be discretized to allow the existence of multiple solution sets in the same level. For this aim, solution set A is superior over solution set B in terms of sparsity if A is sparser than B and the difference between their sparsity is not less than $\frac{1}{D}$, i.e.,

$$sparsity(A) \geq sparsity(B) + \frac{1}{D}. \quad (10)$$

The value $\frac{1}{D}$ indicates the minimum difference between the sparsity of two solution sets. Since $sparsity(A) = sparsity(B) + \frac{1}{D}$ means that each solution in A has one less nonzero variable than each solution in B on average, a difference less than $\frac{1}{D}$ is meaningless and the two solution sets are regarded as having the same sparsity in this case. Then, the solution sets can be assigned to several levels according to their sparsity via the same procedure for assigning the solutions sets according to their dominance relations.

Since the diversity is used to distinguish between the solution sets in the same level, any criterion for diversity assessment can be adopted. While diversity assessment is a challenging issue and existing diversity indicators have more or fewer limitations [26], this work does not aim to suggest a powerful diversity criterion to address the shortcomings of existing indicators. By contrast, the proposed CSD uses a reference vector based diversity criterion for simplicity, where the uniformly distributed reference vectors have been widely used in the environmental selection and diversity assessment of MOEAs [46], [47]. To be specific, each objective vector $\mathbf{f}(\mathbf{x})$ in all the solution sets is first normalized according to the ideal point $\mathbf{z}^*$ and nadir point $\mathbf{z}^{nad}$:

$$f_i(\mathbf{x}) = \frac{f_i(\mathbf{x}) - z_i^*}{z_i^{nad} - z_i^*}, \ i = 1, 2, \ldots, M, \quad (11)$$

where $\mathbf{z}^*$ consists of the minimum objective values in all the solution sets and $\mathbf{z}^{nad}$ consists of the maximum objective values in all the solution sets. Afterwards, a set of uniformly distributed reference vectors R is generated, which has the same size as the solution set having the most solutions. The reference vectors can be generated by the Das and Dennis's method in general, and by the mixture uniform design if the Das and Dennis's method cannot generate the required number of reference vectors [48]. To assess the diversity of each solution set A, each reference vector is associated with the solution closest to it. Then, the number $c_{\mathbf{x}}$ of reference vectors associated with each solution $\mathbf{x}$ is counted:

$$c_{\mathbf{x}} = \left|\{\mathbf{r} \in R | \mathbf{x} = \text{argmin}_{\mathbf{y}\in A} dis(\mathbf{f}(\mathbf{y}), \mathbf{r})\}\right|, \quad (12)$$

where $dis(\mathbf{f}(\mathbf{y}), \mathbf{r})$ denotes the distance between each solution and reference vector, and the standard deviation of all $c_{\mathbf{x}}$ is regarded as the diversity of the solution set:

$$diversity(A) = \sqrt{\frac{1}{|A|}\sum_{\mathbf{x}\in A}(c_{\mathbf{x}} - \bar{c})^2}, \quad (13)$$

where $\bar{c}$ denotes the mean value of all $c_{\mathbf{x}}$.

It is worth noting that the proposed CSD adopts a set of parallel reference vectors rather than a set of reference vectors starting from the origin. As illustrated in Fig. 2, when the seven solutions uniformly distribute on a convex curve, they are associated with $1, 0, 2, 1, 2, 0, 1$ reference vectors starting from the origin, and are associated with $1, 1, 1, 1, 1, 1, 1$ parallel reference vectors. Obviously, the parallel reference vectors can better assess the diversity of solution sets lying on convex or concave surfaces. While the generated reference vectors always start from the origin, the intersection point $\mathbf{r}'$ between each reference

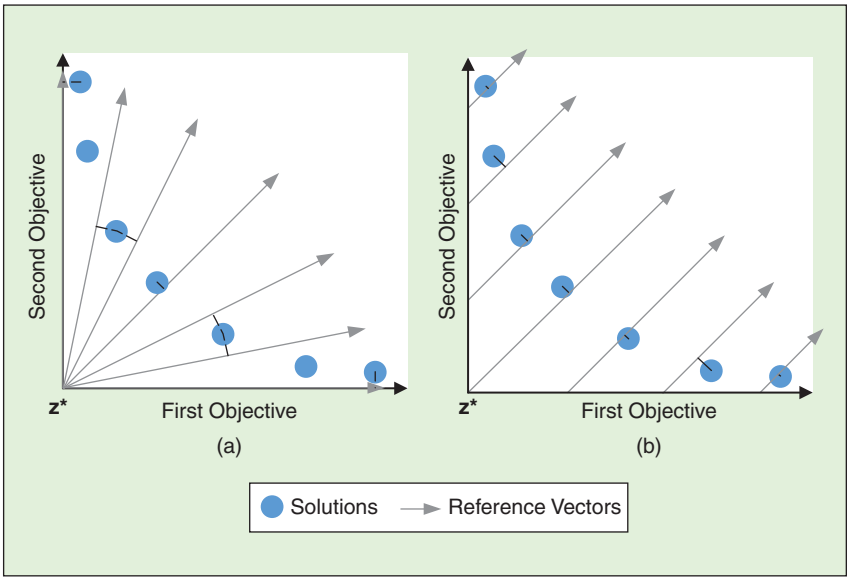


FIGURE 2 Difference between the diversity assessment based on two different sets of reference vectors. (a) $c = (1, 0, 2, 1, 2, 0, 1)$. (b) $c = (1, 1, 1, 1, 1, 1, 1)$.

ALGORITHM 1 Procedure of the proposed CSD.

Input: $A_1, A_2, \ldots$ (solution sets to be compared)
Output: $CSD(A_1), CSD(A_2), \ldots$ (indicator values of the solution sets)
Remove dominated solutions from each solution set temporarily and determine the superiority between each two solution sets in terms of convergence by (4);
$[level^c_1, level^c_2, \ldots] \leftarrow$ Obtain the level of each solution set in terms of convergence;
Determine the superiority between each two solution sets in terms of sparsity by (10);
$[level^s_1, level^s_2, \ldots] \leftarrow$ Obtain the level of each solution set in terms of sparsity;
Normalize the objective vectors of all solutions by (11);
$R \leftarrow$ Generate a set of uniformly distributed reference vectors;
Calculate the intersection points of all reference vectors by (14);
Calculate the foots of all solutions by (15);
$[diversity(A_1), diversity(A_2), \ldots] \leftarrow$ Calculate the diversity of each solution set by (12);
$[CSD(A_1), CSD(A_2), \ldots] \leftarrow$ Calculate the CSD value of each solution set by (17);
return $CSD(A_1), CSD(A_2), \ldots$;

vector $\mathbf{r}$ and the surface $f_1 + f_2 + \cdots + f_M = 1$ is calculated:

$$r_i' = \frac{r_i}{\sum_{j=1}^{M} r_j}, \ i = 1, 2, \ldots, M, \quad (14)$$

and the foot $\mathbf{f}'(\mathbf{x})$ of each solution $\mathbf{x}$ drawn to the surface $f_1 + f_2 + \cdots + f_M = 1$ is calculated:

$$f_i'(\mathbf{x}) = f_i(\mathbf{x}) + \frac{1 - \sum_{j=1}^{M} f_j(\mathbf{x})}{M}, \quad i = 1, 2, \ldots, M. \quad (15)$$

Hence, the distance between each solution and reference vector can be regarded as the Euclidean distance between the foot of the solution and the intersection point of the reference vector.

C. Procedure of the Proposed Indicator

After assigning the solution sets according to their convergence, assigning the solution sets according to their sparsity, and calculating the diversity of the solution set, their CSD values can be obtained by integrating the three criteria. Specifically, the harmonic average of the levels of each solution set in terms of convergence and sparsity is first calculated by

$$ha(A) = \frac{2}{\frac{1}{\text{level}^c_A} + \frac{1}{\text{level}^s_A}}, \quad (16)$$

where $level^c_A$ and $level^s_A$ denote the levels of A in terms of convergence and sparsity, respectively. Then, the ranking $rank_A$ of each solution set A is obtained by sorting all the solution sets according to their harmonic averages of levels. Lastly, the indicator value $CSD(A)$ of each solution set A is calculated by

$$CSD(A) = diversity(A) + \sum_{i=1}^{\text{rank}_A - 1} \max_{\text{rank}_B = i} diversity(B), \quad (17)$$

where $\max_{\text{rank}_B = i} \text{diversity}(B)$ indicates the maximum diversity value among the solution sets with a ranking of i. This way, the indicator values of the solution sets are always larger (i.e., worse) than those with lower rankings. In other words, the indicator value is mainly determined by the convergence and sparsity of a solution set, and the solution sets having the same ranking are distinguished by their diversity. The procedure of the proposed CSD is summarized in Algorithm 1.

As for the computational complexity of the proposed CSD, assuming that the number of solution sets is L, the number of solutions in each set is N, and each solution has M objectives and D variables, the time complexity of assessing their convergence is $O(L^2N^2M)$, the time complexity of assessing their sparsity is $O(LND)$, the time complexity of assessing their diversity is $O(LN^2M)$, and the time complexity of integrating the three criteria is $O(L^2)$. In short, the total time complexity of the proposed CSD is $O(L^2N^2M)$, with the assumption $D \ll LNM$.

D. Discussions

As a matter of fact, most performance indicators assess either convergence or diversity, while only a few can assess both convergence and diversity [49]. Up to now, IGD and HV are still the most practical and popular indicators for the convergence and diversity assessment in multi-objective optimization, whereas no indicator takes the sparsity of solutions into consideration. Thus, the proposed CSD serves as the first indicator for the convergence, sparsity, and diversity assessment in sparse multi-objective optimization. In contrast to IGD and HV, the proposed CSD does not need the true Pareto front or any reference point, thus eliminating the influence of bias introduced by reference points. Moreover, it can be found that the time complexity of the proposed CSD is the same as some indicators (e.g., IGD) while much faster than some others (e.g., HV). As a result, the proposed CSD is practical in real-world scenarios.

It should be noted that the proposed CSD is not Pareto compliant (i.e., non-dominated solutions are always better than dominated ones), since it gives equal priority to convergence and sparsity that may prefer dominated but sparse solutions. Besides, due to the polynary nature of CSD, the indicator values of all the solution sets should be recalculated if a solution set is changed. Nevertheless, it will not consume many additional computational resources since the time complexity of CSD is polynomial rather than exponential.

IV. Experimental and Analysis

This section analyzes the performance of some representative MOEAs on benchmark and real-world large-scale sparse MOPs. To illustrate the effectiveness of the proposed indicator, it is used to assess the quality of the obtained solution sets together with some existing indicators. All the statistical results are obtained by using the open-source platform PlatEMO[1] [53], where the results can be easily reproduced by readers.

A. Experimental settings

1) Algorithms

A total of 11 MOEAs are compared in the experiments, including NSGA-II

[1]https://github.com/BIMK/PlatEMO

TABLE II Detailed settings of the large-scale sparse MOPs used in experiments.

BENCHMARK PROBLEM	TYPE OF VARIABLES	NO. OF VARIABLES	SPARSITY OF PARETO OPTIMAL SOLUTIONS	NO. OF OBJECTIVES		
SMOP1–SMOP8	Real	100 500 1000 5000	0.1	2		
Neural network training	Type of variables	No. of variables	Dataset	No. of samples	No. of features	No. of classes
NN1	Real	301	Wine[1]	178	13	3
NN2		521	Statlog(German)[1]	1000	24	2
NN3		1241	Connectionist Bench Sonar[1]	208	60	2
NN4		6241	LSVT Voice Rehabilitation[1]	126	310	2
Feature selection	Type of variables	No. of variables	Dataset	No. of samples	No. of features	No. of classes
FS1	Binary	100	Hill_Valley[1]	606	100	2
FS2		500	Madelon[1]	2600	500	2
FS3		800	Gse72526[2]	61	800	4
FS4		4434	GLIOMA[3]	50	4434	4
Pattern mining	Type of variables	No. of variables	Dataset	No. of transactions	No. of items	Avg. length of transactions
PM1	Binary	100	Synthetic [50]	1000	100	50
PM2		500	Synthetic [50]	5000	500	250
PM3		1000	Synthetic [50]	10000	1000	500
PM4		5000	Synthetic [50]	50000	5000	2500
Community detection	Type of variables	No. of variables	Dataset	No. of nodes	No. of edges	
CD1	Binary	105	polbooks[4]	105	441	
CD2		453	celegans_metabolic[5]	453	4596	
CD3		1133	Email[5]	1133	5451	
CD4		4039	Facebook [51]	4039	88234	
Critical node detection	Type of variables	No. of variables	Dataset	No. of nodes	No. of edges	
CN1	Binary	102	Movies[6]	102	243	
CN2		500	FF500	500	1078	

TABLE II Detailed settings of the large-scale sparse MOPs used in experiments (Cont.)

BENCHMARK PROBLEM	TYPE OF VARIABLES	NO. OF VARIABLES	SPARSITY OF PARETO OPTIMAL SOLUTIONS	NO. OF OBJECTIVES	
CN3		1000	BA1000	1000	999
CN4		5000	BA5000	5000	4999
Portfolio optimization	Type of variables	No. of variables	Dataset	No. of instruments	Length of each instrument
PO1	Real	100	EURCHF[7]	100	50
PO2		500	EURCHF[7]	500	50
PO3		1000	EURCHF[7]	1000	50
PO4		5000	EURCHF[7]	5000	50
Knapsack problem	Type of variables	No. of variables	Dataset	No. of items	
KP1	Binary	100	Synthetic [52]	100	
KP2		500	Synthetic [52]	500	
KP3		1000	Synthetic [52]	1000	
KP4		5000	Synthetic [52]	5000	

[1]http://archive.ics.uci.edu/ml
[2]https://www.ncbi.nlm.nih.gov/geo/query/acc.cgi
[3]http://featureselection.asu.edu/datasets.php
[4]http://www-personal.umich.edu/%7emejn/netdata/
[5]http://deim.urv.cat/%7ealexandre.arenas/data/welcome.htm
[6]http://vlado.fmf.uni-lj.si/pub/networks/data/default.htm
[7]https://www.metatrader5.com/en

[29], MOEA/D-DE [30], MOEA/DVA [5], LMEA [6], WOF-SMPSO [7], LSMOF-NSGA-II [8], IM-MOEA [32], LMOCSO [11], SparseEA [20], MOEA/PSL [22], and PM-MOEA [27]. NSGA-II and MOEA/D-DE stand for classical MOEAs, which generate offspring solutions by using genetic operators and differential evolution, respectively. The next six MOEAs are state-of-the-art algorithms for solving large-scale MOPs, where MOEA/DVA and LMEA are based on variable grouping, WOF-SMPSO and LSMOF-NSGA-II are based on dimensionality reduction, and IM-MOEA and LMOCSO are based on novel variation operators. The last three MOEAs are tailored for large-scale sparse MOPs, which search for sparse Pareto optimal solutions by using bi-level encoding, unsupervised neural networks, and evolutionary pattern mining, respectively. For MOEA/D-DE, the neighborhood size is set to 10, the probability of choosing parents locally is set to 0.9, and the maximum number of solutions replaced by each offspring solution is set to 2. For MOEA/DVA, the number of sampling solutions in variable analysis is set to 20 and the number of selected solutions for variable interaction analysis is set to 5. For LMEA, the number of selected solutions for variable clustering is set to 2, the number of perturbations on each solution for variable clustering is set to 4, and the number of selected solutions for variable interaction analysis is set to 5. For WOF-SMPSO, the number of groups is set to 4, the number of evaluations for the original problem is set to 1000, the number of evaluations for the transformed problem is set to 500, the number of chosen solutions for weight optimization is set to 3, and the fraction of evaluations for weight optimization is set to 0.5. For LSMOF-NSGA-II, the number of reference solutions is set to 10 and the population size of the transformed problem is set to 30. For IM-MOEA, the number of reference vectors is set to 10 and the model group size is set to 3.

2) Operators

To generate offspring solutions with real variables, NSGA-II, LMEA, LSMOF-NSGA-II, SparseEA, MOEA/PSL, and PM-MOEA use simulated binary crossover (SBX) [54] and polynomial mutation (PM) [55], MOEA/DVA and MOEA/D-DE use differential evolution and PM, WOF-SMPSO uses particle swarm optimization and PM, IM-MOEA uses a Gaussian process based inverse model and PM, and LMOCSO uses an enhanced competitive swarm optimizer and PM. To generate offspring solutions with binary variables, NSGA-II and MOEA/PSL use uniform crossover and bit-flip mutation, SparseEA and PM-MOEA use their own crossover and mutation operators. Besides, MOEA/D-DE, MOEA/DVA, LMEA, WOF-SMPSO, LSMOF-NSGA-II, IM-MOEA, and LMOCSO generate real variables within $[0,1]^D$ and round them to obtain binary solutions. The distribution index of SBX and PM is set to 20, the probability of SBX and uniform crossover is set to 1, the probability of PM and bit-flip mutation is set to $\frac{1}{D}$, and the parameters CR and F in differential evolution are set to 1 and 0.5, respectively.

3) Number of evaluations and population size

For fair comparisons, the number of function evaluations available to all the MOEAs is set to $100 \times D$, and the population size of all the MOEAs is set to 100.

4) Problems

The MOEAs are compared on eight benchmark problems (i.e., SMOP1–SMOP8) and seven real-world applications (i.e., neural network training, feature selection, pattern mining, community detection, critical node detection, portfolio optimization, and the knapsack problem). The definitions of the eight benchmark problems can be found in [20], the definitions of the first six real-world applications can be found in [22], and the definition of the knapsack problem is referred to [52]. For the knapsack problem, the capacity of each knapsack is set to one tenth of the total weight of all items, and the constraint violations (i.e., caused by overweight items) are added to the objectives to avoid explicit constraints. For all the problems, the number of objectives is set to 2, the number of decision variables varies from 100 to 5000, and the other parameter settings are listed in Table II.

B. Comparisons on Benchmark Problems

Table III lists the CSD values of the solution sets obtained by the compared MOEAs on benchmark problems SMOP1–SMOP8 with 100 to 5000 decision variables, averaged over 30 runs. The best CSD value in each row is shown in bold, and all the results statistically similar to the best one in terms of the Wilcoxon rank sum test [56] with a significance level of 0.05 are highlighted. According to the table, five observations can be made: Firstly, genetic algorithms have better average rankings than differential evolution (i.e., NSGA-II vs. MOEA/D-DE and LMEA vs. MOEA/DVA), which is mainly due to the powerful exploration ability of genetic operators on multimodal landscapes. Secondly, the variable grouping based MOEAs (i.e., MOEA/DVA and LMEA) and novel variation operator based MOEAs (i.e., IM-MOEA and LMOCSO) exhibit worse performance than classical MOEAs (i.e., NSGA-II and MOEA/D-DE), which indicates that these

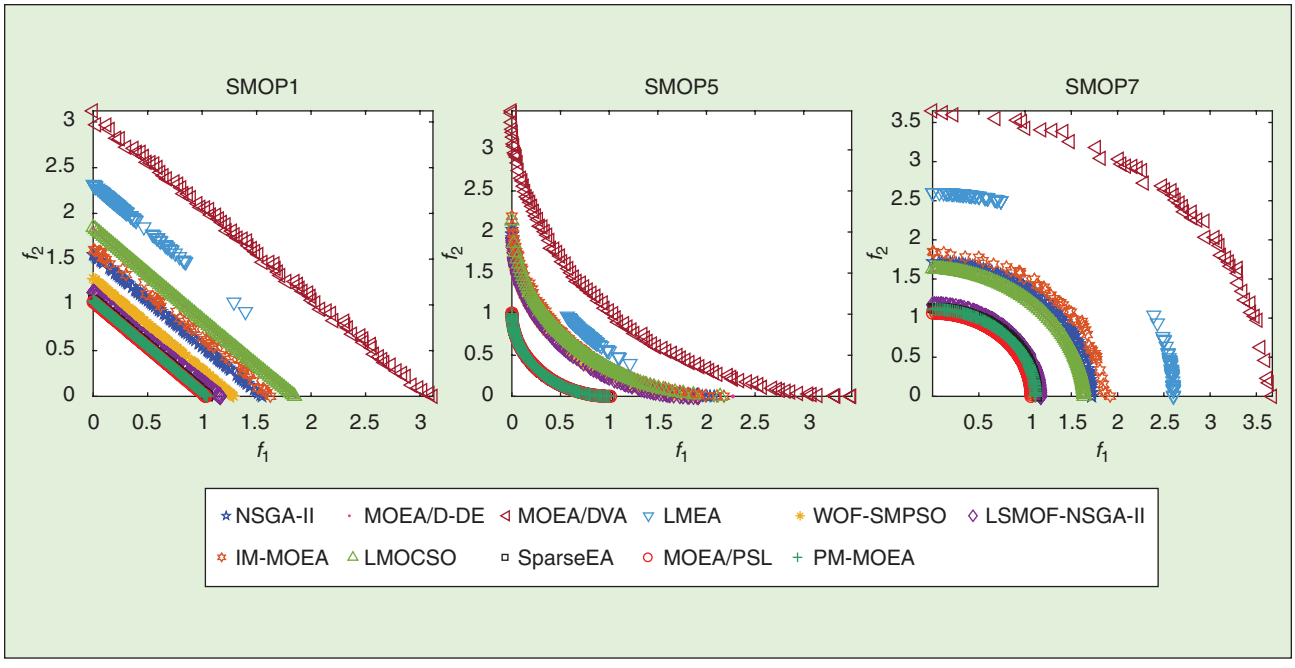


FIGURE 3 Solution sets (in objective space) with median CSD obtained by 11 MOEAs on SMOP1, SMOP5, and SMOP7 with 5000 decision variables.

TABLE III CSD values obtained by 11 MOEAs on SMOP1–SMOP8 with 100 to 5000 decision variables.

PROBLEM (D)	NSGA-II	MOEA/D-DE	MOEA/DVA	LMEA	WOF-SMPSO	LSMOF-NSGA-II	IM-MOEA	LMOCSO	SparseEA	MOEA/PSL	PM-MOEA
SMOP1 (100)	1.9337e-1	4.3614e-1	7.8851e-1	5.9369e-1	2.3918e-1	1.4151e-1	3.7767e-1	4.1068e-1	6.9883e-2	**4.8689e-2**	9.4315e-2
SMOP2 (100)	3.0427e-1	6.8635e-1	8.1987e-1	6.7602e-1	3.7089e-1	2.4525e-1	5.4250e-1	6.5158e-1	1.3044e-1	**5.5829e-2**	9.0096e-2
SMOP3 (100)	2.9980e-1	3.3553e-1	6.9653e-1	4.9054e-1	2.5190e-1	2.0252e-1	5.7921e-1	3.5256e-1	8.6080e-2	**7.0560e-2**	1.0548e-1
SMOP4 (100)	1.6997e-1	6.4059e-1	7.5967e-1	5.4760e-1	3.3175e-1	1.1802e-1	3.4479e-1	5.2913e-1	6.5248e-2	**4.9980e-2**	5.6740e-2
SMOP5 (100)	1.4063e-1	3.7059e-1	6.1990e-1	4.6552e-1	1.4168e-1	9.6309e-2	3.0165e-1	2.4447e-1	4.6993e-2	4.4203e-2	**3.7296e-2**
SMOP6 (100)	1.3167e-1	4.6350e-1	7.2521e-1	6.0050e-1	1.7961e-1	8.8620e-2	2.7609e-1	3.4374e-1	**4.4372e-2**	4.7910e-2	4.8082e-2
SMOP7 (100)	7.6009e-1	8.7693e-1	1.6762e+0	1.3955e+0	4.1530e-1	6.1877e-1	1.1281e+0	9.0127e-1	2.1786e-1	**1.0306e-1**	2.4001e-1
SMOP8 (100)	1.0803e+0	1.3295e+0	1.8521e+0	1.6576e+0	7.8605e-1	8.5502e-1	1.4568e+0	1.2191e+0	3.2522e-1	**1.5662e-1**	2.9235e-1
SMOP1 (500)	2.0078e-1	4.7565e-1	6.5680e-1	5.2181e-1	2.5840e-1	1.4826e-1	3.8710e-1	4.1421e-1	7.7962e-2	**6.8524e-2**	7.4613e-2
SMOP2 (500)	2.2361e-1	5.3837e-1	6.2014e-1	4.9237e-1	1.7649e-1	1.3007e-1	4.0178e-1	5.1778e-1	6.1991e-2	**4.6055e-2**	7.5849e-2
SMOP3 (500)	2.1226e-1	2.2364e-1	4.5775e-1	3.1473e-1	1.6350e-1	1.1170e-1	3.8878e-1	2.2364e-1	5.4376e-2	**5.0797e-2**	5.1477e-2
SMOP4 (500)	2.9639e-1	7.4486e-1	8.4093e-1	5.4238e-1	2.4601e-1	1.9151e-1	4.1929e-1	6.3857e-1	9.1078e-2	**6.0793e-2**	1.3729e-1
SMOP5 (500)	1.5555e-1	3.6846e-1	6.8601e-1	5.8986e-1	1.0433e-1	1.5748e-1	4.2196e-1	2.7542e-1	4.7540e-2	5.6343e-2	**4.0170e-2**
SMOP6 (500)	1.3524e-1	4.9684e-1	5.8008e-1	4.4997e-1	1.8761e-1	8.2919e-2	2.7406e-1	3.5664e-1	5.9739e-2	4.7825e-2	**3.3376e-2**
SMOP7 (500)	2.9905e-1	3.5295e-1	9.4878e-1	7.4620e-1	1.6603e-1	2.4426e-1	5.9310e-1	4.1848e-1	8.1813e-2	**5.2182e-2**	1.0454e-1
SMOP8 (500)	3.6262e-1	5.0656e-1	9.3498e-1	7.8791e-1	1.8598e-1	3.0491e-1	6.5754e-1	4.3061e-1	1.0165e-1	**7.5297e-2**	8.8172e-2
SMOP1 (1000)	1.5571e-1	3.5225e-1	5.2676e-1	4.3443e-1	1.9660e-1	1.0731e-1	3.1717e-1	3.3554e-1	**4.8398e-2**	5.3307e-2	5.3906e-2
SMOP2 (1000)	2.5495e-1	5.2097e-1	5.9670e-1	4.9781e-1	2.0615e-1	1.5315e-1	4.0175e-1	5.3552e-1	9.2543e-2	7.0190e-2	**4.0488e-2**
SMOP3 (1000)	2.5065e-1	2.5065e-1	4.6366e-1	3.4697e-1	1.9917e-1	1.4570e-1	4.0951e-1	2.5065e-1	7.5598e-2	6.2881e-2	**5.5213e-2**
SMOP4 (1000)	2.9679e-1	7.9097e-1	8.7980e-1	5.8962e-1	2.4536e-1	1.9047e-1	4.0549e-1	6.9149e-1	1.0773e-1	**5.3168e-2**	1.4350e-1
SMOP5 (1000)	1.8111e-1	3.9310e-1	5.8350e-1	5.9136e-1	1.2368e-1	1.4881e-1	4.1826e-1	2.8160e-1	4.8329e-2	7.0232e-2	**3.6738e-2**
SMOP6 (1000)	1.3829e-1	4.9495e-1	5.5872e-1	5.1353e-1	1.9575e-1	8.9246e-2	2.7406e-1	3.5636e-1	5.7827e-2	5.0686e-2	**3.5731e-2**
SMOP7 (1000)	3.0360e-1	3.5633e-1	9.0563e-1	7.2155e-1	1.7128e-1	2.4907e-1	5.8356e-1	4.1170e-1	6.9131e-2	**6.6191e-2**	1.1651e-1
SMOP8 (1000)	3.6972e-1	5.6807e-1	9.0280e-1	7.3794e-1	1.7119e-1	3.0418e-1	7.2364e-1	4.2391e-1	8.5294e-2	**5.5697e-2**	7.7375e-2
SMOP1 (5000)	3.0581e-1	4.3757e-1	7.1662e-1	6.5268e-1	2.5393e-1	2.0094e-1	4.3473e-1	4.3757e-1	1.1666e-1	8.4255e-2	**7.2000e-2**
SMOP2 (5000)	2.3548e-1	3.9400e-1	4.1265e-1	5.3216e-1	1.3689e-1	1.8393e-1	3.8831e-1	3.9517e-1	6.6315e-2	**5.1404e-2**	6.6043e-2
SMOP3 (5000)	2.0494e-1	2.0494e-1	4.0782e-1	3.3496e-1	1.5823e-1	1.6087e-1	3.9645e-1	2.0494e-1	8.5894e-2	**4.6221e-2**	6.6826e-2
SMOP4 (5000)	2.9513e-1	6.3529e-1	7.4225e-1	6.3870e-1	2.0108e-1	2.4033e-1	3.9630e-1	5.7324e-1	8.9054e-2	**7.9108e-2**	1.4608e-1
SMOP5 (5000)	1.8998e-1	3.9632e-1	4.8896e-1	6.4584e-1	1.3207e-1	1.3989e-1	4.4364e-1	2.8988e-1	4.8379e-2	8.3544e-2	**4.6288e-2**
SMOP6 (5000)	2.1177e-1	4.6564e-1	5.0253e-1	7.2834e-1	1.4962e-1	1.2542e-1	2.8417e-1	3.6945e-1	6.1343e-2	6.5937e-2	**3.5237e-2**
SMOP7 (5000)	3.7257e-1	2.9600e-1	1.0383e+0	9.1123e-1	1.5326e-1	2.3453e-1	5.0184e-1	3.1529e-1	1.1192e-1	**7.0545e-2**	1.1669e-1
SMOP8 (5000)	2.7640e-1	4.3124e-1	1.0415e+0	9.3653e-1	1.7614e-1	2.2673e-1	5.5521e-1	3.2799e-1	1.1516e-1	7.5387e-2	**4.4007e-2**
Average ranking	5.7500	8.3125	10.6875	9.4688	4.8438	4.4688	7.9375	7.7500	2.4375	1.5313	2.0313

TABLE IV IGD values obtained by 11 MOEAs on SMOP1–SMOP8 with 100 to 5000 decision variables.

PROBLEM (D)	NSGA-II	MOEA/D-DE	MOEA/DVA	LMEA	WOF-SMPSO	LSMOF-NSGA-II	IM-MOEA	LMOCSO	SparseEA	MOEA/PSL	PM-MOEA
SMOP1 (100)	1.3881e-1	4.9338e-1	1.1699e+0	7.5073e-1	2.7445e-1	5.4056e-2	3.7859e-1	4.5991e-1	9.6493e-3	**7.5149e-3**	1.0619e-2
SMOP2 (100)	5.9326e-1	1.6633e+0	1.9698e+0	1.6498e+0	9.4523e-1	1.4431e-1	1.2576e+0	1.6623e+0	3.7669e-2	**1.8031e-2**	2.4476e-2
SMOP3 (100)	8.9196e-1	1.5554e+0	2.3901e+0	1.9913e+0	7.2635e-1	7.1078e-1	2.1896e+0	1.6021e+0	1.6880e-2	**1.2013e-2**	1.3663e-2
SMOP4 (100)	1.8464e-1	8.2157e-1	1.0420e+0	8.0055e-1	5.2006e-1	6.4901e-3	4.7911e-1	8.0353e-1	5.0302e-3	4.6541e-3	**4.1171e-3**
SMOP5 (100)	3.7593e-1	4.2447e-1	8.2484e-1	5.6683e-1	3.7125e-1	3.6417e-1	4.2339e-1	4.1629e-1	6.0333e-3	6.7538e-3	**5.8328e-3**
SMOP6 (100)	5.1558e-2	1.7537e-1	3.7933e-1	2.2736e-1	8.3967e-2	8.2906e-3	1.4209e-1	1.5128e-1	8.3891e-3	**8.0869e-3**	9.3224e-3
SMOP7 (100)	3.7936e-1	4.8911e-1	1.9933e+0	1.5488e+0	1.7419e-1	2.2770e-1	7.6863e-1	5.4583e-1	3.7442e-2	**2.8884e-2**	4.2378e-2
SMOP8 (100)	1.6116e+0	2.6267e+0	3.3817e+0	3.2100e+0	8.1791e-1	7.0334e-1	2.6644e+0	2.4326e+0	1.6878e-1	**1.5041e-1**	1.7536e-1
SMOP1 (500)	1.9086e-1	5.9608e-1	1.3679e+0	5.8926e-1	2.6890e-1	5.1330e-2	4.4007e-1	5.3002e-1	1.9205e-2	**1.1723e-2**	1.6715e-2
SMOP2 (500)	8.1616e-1	1.8533e+0	2.0746e+0	1.4881e+0	3.1368e-1	1.2903e-1	1.1803e+0	1.7608e+0	5.9364e-2	**2.3902e-2**	3.6876e-2
SMOP3 (500)	1.0870e+0	1.6620e+0	2.4751e+0	1.7723e+0	7.0546e-1	6.8492e-1	2.3002e+0	1.6474e+0	2.2764e-2	**1.1323e-2**	2.0308e-2
SMOP4 (500)	3.4758e-1	9.1579e-1	1.0823e+0	7.0549e-1	8.9249e-2	4.8261e-3	6.1920e-1	8.5216e-1	5.0911e-3	4.6319e-3	**4.0797e-3**
SMOP5 (500)	3.7684e-1	4.3490e-1	9.5036e-1	5.0357e-1	3.5916e-1	3.7463e-1	4.4195e-1	4.2999e-1	6.1487e-3	7.1022e-3	**5.2572e-3**
SMOP6 (500)	5.4521e-2	2.0103e-1	4.1971e-1	1.8076e-1	7.9325e-2	6.3450e-3	1.6410e-1	1.8067e-1	7.6552e-3	7.0130e-3	**5.0900e-3**
SMOP7 (500)	4.0898e-1	5.3185e-1	2.3711e+0	1.1668e+0	9.3594e-2	1.7344e-1	8.3552e-1	6.2655e-1	6.4013e-2	**4.3760e-2**	5.8032e-2
SMOP8 (500)	1.7140e+0	2.6506e+0	3.5499e+0	3.0276e+0	5.6815e-1	7.1543e-1	2.8422e+0	2.3489e+0	2.3051e-1	**1.8299e-1**	1.7838e-1
SMOP1 (1000)	2.3802e-1	6.0626e-1	1.4011e+0	6.3298e-1	2.5378e-1	5.8896e-2	4.3913e-1	5.5384e-1	2.5844e-2	**1.5394e-2**	2.1628e-2
SMOP2 (1000)	9.5843e-1	1.8104e+0	2.1029e+0	1.4979e+0	4.4390e-1	1.3484e-1	1.2829e+0	1.8253e+0	7.1310e-2	**4.1015e-2**	5.2607e-2
SMOP3 (1000)	1.2041e+0	1.6637e+0	2.4833e+0	1.8615e+0	7.0493e-1	6.8275e-1	2.3367e+0	1.6860e+0	2.8652e-2	**1.4260e-2**	2.4684e-2
SMOP4 (1000)	4.2098e-1	9.4218e-1	1.0962e+0	7.0554e-1	5.2047e-2	4.8741e-3	6.2804e-1	8.8435e-1	4.9091e-3	4.6623e-3	**4.1184e-3**
SMOP5 (1000)	3.8407e-1	4.4126e-1	9.5534e-1	5.0427e-1	3.5629e-1	3.6945e-1	4.4841e-1	4.3506e-1	5.7937e-3	7.2164e-3	**4.7038e-3**
SMOP6 (1000)	6.4852e-2	2.0711e-1	4.2930e-1	1.9409e-1	8.2004e-2	6.1533e-3	1.6891e-1	1.8522e-1	7.5569e-3	7.2044e-3	**4.9861e-3**
SMOP7 (1000)	4.7519e-1	5.4254e-1	2.4557e+0	1.1352e+0	9.3192e-2	1.8547e-1	8.2090e-1	6.1863e-1	8.2947e-2	**5.2677e-2**	7.3822e-2
SMOP8 (1000)	1.9015e+0	2.7501e+0	3.5972e+0	2.8570e+0	5.6434e-1	7.1966e-1	2.6160e+0	2.3568e+0	2.5142e-1	**1.7740e-1**	1.8902e-1
SMOP1 (5000)	4.1609e-1	5.6506e-1	1.4404e+0	9.4475e-1	1.8574e-1	1.0086e-1	4.4519e-1	5.9038e-1	3.8444e-2	**1.8127e-2**	3.4955e-2
SMOP2 (5000)	1.2129e+0	1.7842e+0	2.1319e+0	1.7309e+0	1.7927e-1	2.0046e-1	1.3522e+0	1.7897e+0	9.8401e-2	**5.5943e-2**	8.7136e-2
SMOP3 (5000)	1.6060e+0	1.6850e+0	2.5065e+0	2.1669e+0	7.0202e-1	7.0319e-1	2.3627e+0	1.7051e+0	4.5589e-2	**1.3926e-2**	3.2480e-2
SMOP4 (5000)	5.8178e-1	9.1493e-1	1.1102e+0	8.1680e-1	9.6085e-3	1.3846e-2	6.8302e-1	9.0018e-1	4.9284e-3	4.7285e-3	**4.0531e-3**
SMOP5 (5000)	4.2033e-1	4.5025e-1	9.7775e-1	6.3210e-1	3.4951e-1	3.5416e-1	4.5318e-1	4.4112e-1	5.7860e-3	7.0693e-3	**4.8330e-3**
SMOP6 (5000)	1.2156e-1	2.1264e-1	4.4097e-1	2.7834e-1	2.7805e-2	1.8836e-2	1.7937e-1	1.9077e-1	8.1920e-3	7.8950e-3	**5.3264e-3**
SMOP7 (5000)	7.5066e-1	5.7751e-1	2.5780e+0	1.5629e+0	8.4597e-2	1.7858e-1	8.4136e-1	6.1830e-1	1.2622e-1	**7.6508e-2**	1.1568e-1
SMOP8 (5000)	2.3367e+0	2.7799e+0	3.6356e+0	3.0393e+0	5.3245e-1	5.6989e-1	2.7928e+0	2.4849e+0	3.2451e-1	**2.0764e-1**	2.9041e-1
Average ranking	5.8125	8.5625	11.0000	9.2813	4.8125	4.1563	8.0313	8.0313	2.9375	1.5625	1.8125

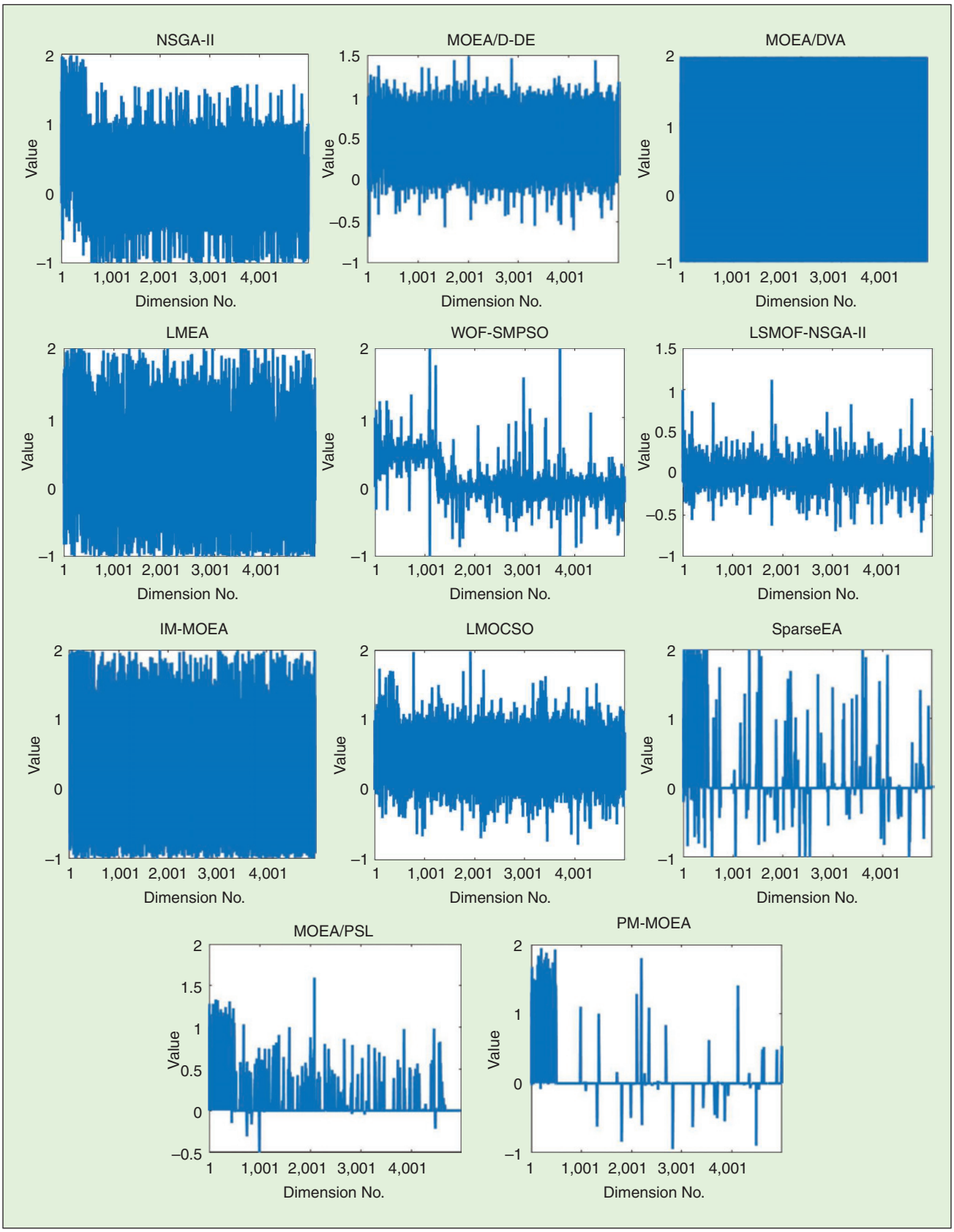


FIGURE 4 Parallel coordinates of solution sets (in decision space) with median CSD obtained by 11 MOEAs on SMOP1 with 5000 decision variables.

TABLE V CSD values obtained by 11 MOEAs on seven applications with 100 to 6241 decision variables.

PROBLEM (D)	NSGA-II	MOEA/D-DE	MOEA/DVA	LMEA	WOF-SMPSO	LSMOF-NSGA-II	IM-MOEA	LMOCSO	SparseEA	MOEA/PSL	PM-MOEA
NN1 (301)	3.0048e+0	2.2764e+0	4.8785e+0	1.7258e+0	3.7028e+0	3.2312e+0	2.2574e+0	1.5509e+0	5.3702e-1	**4.1100e-1**	7.1119e-1
NN2 (521)	2.7123e+0	2.6272e+0	5.1884e+0	3.0637e+0	3.9284e+0	3.9938e+0	2.7048e+0	1.8528e+0	6.8162e-1	**6.2156e-1**	8.1119e-1
NN3 (1241)	2.7509e+0	2.3780e+0	4.9408e+0	3.2701e+0	3.5023e+0	3.5899e+0	1.7052e+0	1.7826e+0	**2.8771e-1**	7.6494e-1	8.1605e-1
NN4 (6241)	2.8317e+0	2.3007e+0	5.0684e+0	3.3388e+0	3.0731e+0	3.8022e+0	2.2865e+0	1.9593e+0	**2.1026e-1**	9.7486e-1	8.5385e-1
FS1 (100)	1.0310e+0	1.8985e+0	4.2886e+0	2.0280e+0	1.2903e+0	1.0999e+0	3.1714e+0	2.9045e+0	6.6861e-1	**5.7484e-1**	6.2380e-1
FS2 (500)	1.9188e+0	2.5088e+0	5.1181e+0	2.6017e+0	1.3877e+0	7.6036e-1	3.9152e+0	3.7789e+0	**5.4259e-1**	5.5087e-1	8.3448e-1
FS3 (800)	7.4050e-1	1.5696e+0	5.6488e+0	2.8488e+0	4.2424e-1	**3.2308e-1**	4.4488e+0	4.0488e+0	**3.2308e-1**	4.3138e-1	6.4616e-1
FS4 (4434)	9.6118e-1	1.6380e+0	5.6669e+0	2.8669e+0	5.0166e-1	**2.7824e-1**	4.4669e+0	4.0669e+0	3.8634e-1	6.8423e-1	7.7268e-1
PM1 (100)	7.6220e-1	3.6105e+0	5.0103e+0	1.1951e+0	3.4305e+0	3.1172e+0	2.7137e+0	3.2038e+0	**6.3366e-1**	1.5542e+0	1.1475e+0
PM2 (500)	**4.6142e-1**	2.3554e+0	4.5193e+0	8.6373e-1	3.4327e+0	3.0287e+0	1.9572e+0	2.7711e+0	7.0696e-1	1.6241e+0	1.2144e+0
PM3 (1000)	5.7829e-1	3.2521e+0	5.2072e+0	9.7103e-1	3.8133e+0	3.6349e+0	2.5700e+0	3.3428e+0	1.6347e+0	1.7843e+0	**5.4839e-1**
PM4 (5000)	1.3943e+0	3.6887e+0	5.3115e+0	9.9924e-1	3.7022e+0	3.7936e+0	2.6837e+0	2.2026e+0	2.0138e+0	1.1828e+0	**6.0856e-1**
CD1 (105)	5.3290e-1	1.8945e+0	2.6848e+0	1.0816e+0	1.2997e+0	1.3088e+0	1.1181e+0	2.0159e+0	5.5563e-1	3.1490e-1	**2.1683e-1**
CD2 (453)	9.0566e-1	2.8390e+0	3.8227e+0	1.5851e+0	2.1970e+0	2.0061e+0	1.9279e+0	2.8943e+0	5.6652e-1	**4.1858e-1**	8.0706e-1
CD3 (1133)	7.2943e-1	2.3132e+0	3.5010e+0	1.4050e+0	1.3378e+0	1.5063e+0	1.7765e+0	2.7489e+0	3.6724e-1	5.2786e-1	**3.5349e-1**
CD4 (4039)	8.4893e-1	2.7008e+0	3.6898e+0	1.7421e+0	1.5173e+0	1.7041e+0	2.0949e+0	2.8086e+0	3.8743e-1	9.5351e-1	**2.9303e-1**
CN1 (102)	3.0989e-1	1.7920e+0	2.6261e+0	1.0605e+0	1.0771e+0	1.3401e+0	1.1292e+0	1.9874e+0	6.3345e-1	3.9408e-1	**1.7191e-1**
CN2 (500)	4.2488e-1	2.2900e+0	3.2954e+0	1.4615e+0	1.0587e+0	1.9627e+0	1.6843e+0	2.3819e+0	6.7480e-1	**2.4352e-1**	1.0404e+0
CN3 (1000)	8.4086e-1	3.0956e+0	3.7315e+0	1.8393e+0	1.5798e+0	2.3397e+0	2.3631e+0	2.8499e+0	8.6170e-1	6.2590e-1	**3.8083e-1**
CN4 (5000)	9.4161e-1	3.5873e+0	4.3802e+0	2.2234e+0	1.6530e+0	2.5253e+0	2.8313e+0	3.5155e+0	7.6144e-1	9.0327e-1	**6.6124e-1**
PO1 (100)	2.1728e+0	4.4328e+0	6.1017e+0	4.5555e+0	3.4440e+0	4.8379e+0	4.5704e+0	4.2169e+0	1.8698e+0	**6.5916e-1**	1.6701e+0
PO2 (500)	1.8878e+0	4.1223e+0	5.4235e+0	3.2365e+0	2.6118e+0	4.0805e+0	4.4928e+0	4.1498e+0	1.3542e+0	**7.4433e-1**	2.8456e+0
PO3 (1000)	2.1851e+0	4.0712e+0	5.1581e+0	3.3181e+0	2.3105e+0	4.0289e+0	4.0696e+0	3.7363e+0	**7.4388e-1**	1.0596e+0	1.3492e+0
PO4 (5000)	2.2480e+0	4.3841e+0	4.9446e+0	3.4730e+0	2.4696e+0	3.3497e+0	3.8710e+0	4.1257e+0	**8.9096e-1**	1.4762e+0	1.6541e+0
KP1 (100)	2.3645e+0	6.9903e+0	8.1903e+0	3.7214e+0	4.5562e+0	3.5374e+0	6.4506e+0	5.2506e+0	1.5292e+0	**8.3318e-1**	1.2534e+0
KP2 (500)	1.5385e+0	7.0932e+0	5.6739e+0	4.6544e+0	3.0759e+0	2.6566e+0	6.3339e+0	3.7750e+0	9.7928e-1	**9.7818e-1**	1.8173e+0
KP3 (1000)	1.5456e+0	6.9999e+0	5.7187e+0	4.1731e+0	2.6701e+0	4.8160e+0	7.0592e+0	3.7728e+0	1.1240e+0	**7.0271e-1**	1.9674e+0
KP4 (5000)	1.8292e+0	5.8212e+0	4.8805e+0	3.7548e+0	2.6734e+0	5.3619e+0	6.7026e+0	4.1176e+0	9.8494e-1	**7.0353e-1**	1.6885e+0
Average ranking	4.0357	8.3214	10.7143	6.1786	6.3571	7.1429	7.8571	7.8214	2.2857	2.2857	2.6429

search strategies for tackling high-dimensional decision spaces are inefficient in finding sparse solutions. Thirdly, the dimensionality reduction based MOEAs (i.e., WOF-SMPSO and LSMOF-NSGA-II) considerably outperform the MOEAs based on variable grouping and novel variation operators, since the former can quickly converge to quasi-optimal regions but the latter cannot converge by using $100 \times D$ function evaluations. In particular, LMOCSO is verified to be effective with $15000 \times D$ function evaluations [11], which is unaffordable for real-world applications.

Fourthly, the sparse MOEAs (i.e., SparseEA, MOEA/PSL, and PM-MOEA) obtain the best CSD values and have obviously better rankings than the other MOEAs, where the effectiveness of these MOEAs in searching for sparse Pareto optimal solutions can be confirmed. Fig. 3 depicts the objective values of the solution sets with median CSD obtained by the compared MOEAs on SMOP1, SMOP6, and SMOP7 with 5000 decision variables, where the solution sets obtained by SparseEA, MOEA/PSL, and PM-MOEA have better convergence than those obtained by classical MOEAs and large-scale MOEAs. Lastly, although all of the three sparse MOEAs maintain the sparsity of solutions by using the bi-level encoding in (3), PM-MOEA outperforms SparseEA while MOEA/PSL outperforms both SparseEA and PM-MOEA. This is because SparseEA does not adopt any dimensionality reduction strategies, while PM-MOEA uses an evolutionary pattern mining approach to reduce the dimensionality of the binary vector, and MOEA/PSL uses two unsupervised neural networks to reduce the dimensionality of both the binary vector and real vector.

To verify the effectiveness of the proposed indicator, Table IV presents the IGD values obtained by the compared MOEAs, where 10000 reference points are sampled by the method suggested in [42] for IGD calculation. It can be observed from the table that the average rankings of the compared MOEAs are similar to those in Table III, where MOEA/PSL obtains the best performance, PM-MOEA obtains the second best performance, and SparseEA obtains the third best performance. Nonetheless, the IGD values of some results are inconsistent with the corresponding CSD values in Table III. For example, PM-MOEA obtains better CSD value than MOEA/PSL on SMOP1 with 5000 decision variables, while MOEA/PSL gains better IGD value than PM-MOEA on the same test instance. According to the decision variables of the solution sets obtained for SMOP1 with 5000 decision variables shown in Fig. 4, the solution set obtained by PM-MOEA is sparser than that obtained by MOEA/PSL. Hence, PM-MOEA obtains the best CSD value since it can find very sparse solutions with the assistance of the evolutionary pattern mining approach, and the proposed CSD considers the sparsity of solution sets. By contrast, MOEA/PSL gains the best IGD value since it can better optimize the nonzero variables by using the restricted Boltzmann machine, and IGD does not take the sparsity of solution sets into consideration. As a consequence, the proposed CSD is more effective than IGD in assessing the solution sets for sparse MOPs.

C. Comparisons on Real-World SMOPs

Next, the performance of the 11 MOEAs is compared on seven real-world applications with 100 to 6241 decision variables. These real-world problems hold some characteristics

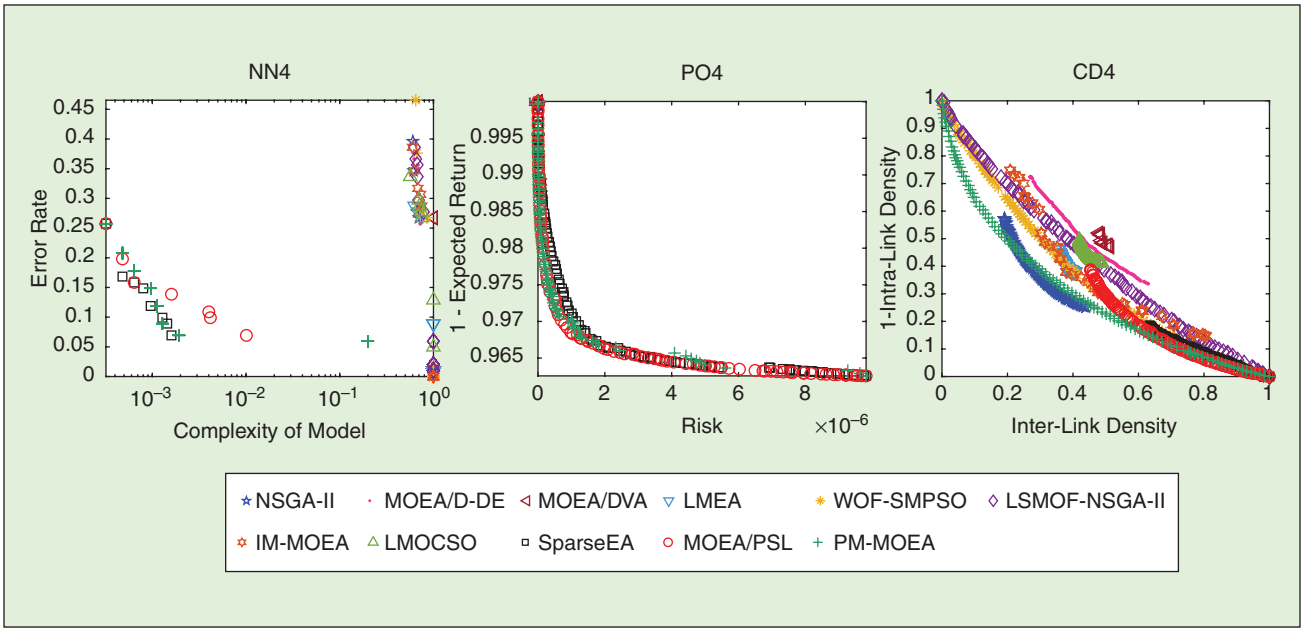


FIGURE 5 Solution sets (in objective space) with median CSD obtained by 11 MOEAs on neural network training, portfolio optimization, and community detection with approximately 5000 decision variables.

TABLE VI HV values obtained by 11 MOEAs on seven applications with 100 to 6241 decision variables.

PROBLEM (D)	NSGA-II	MOEA/D-DE	MOEA/DVA	LMEA	WOF-SMPSO	LSMOF-NSGA-II	IM-MOEA	LMOCSO	SparseEA	MOEA/PSL	PM-MOEA
NN1 (301)	4.0033e-1	3.9943e-1	8.2934e-2	4.1292e-1	3.9293e-1	3.9402e-1	3.9730e-1	3.9862e-1	9.9175e-1	**9.9425e-1**	9.7726e-1
NN2 (521)	3.0025e-1	3.0287e-1	5.9979e-2	3.0456e-1	3.0817e-1	3.0323e-1	3.0944e-1	2.9797e-1	7.9986e-1	**8.0317e-1**	7.7881e-1
NN3 (1241)	3.4234e-1	3.4782e-1	6.8343e-2	3.4964e-1	3.6260e-1	3.4953e-1	3.5777e-1	3.5647e-1	**8.9675e-1**	8.6952e-1	8.6733e-1
NN4 (6241)	3.5159e-1	3.4005e-1	6.7343e-2	3.4481e-1	3.3685e-1	3.4354e-1	3.5219e-1	3.6199e-1	9.2786e-1	9.3084e-1	**9.3878e-1**
FS1 (100)	7.5146e-1	5.7736e-1	5.9490e-2	5.8842e-1	7.2928e-1	7.1699e-1	4.7300e-1	4.7016e-1	7.3136e-1	7.4621e-1	**7.5187e-1**
FS2 (500)	6.9445e-1	5.5587e-1	5.4355e-2	5.6351e-1	9.1048e-1	9.0097e-1	4.3429e-1	4.2526e-1	**9.2304e-1**	9.1924e-1	9.1365e-1
FS3 (800)	9.9902e-1	4.8129e-1	3.5813e-2	3.2751e-1	9.9907e-1	**9.9914e-1**	2.3090e-1	2.3153e-1	**9.9914e-1**	9.9905e-1	**9.9914e-1**
FS4 (4434)	9.9960e-1	6.8766e-1	4.1322e-2	3.6185e-1	9.6340e-1	9.9974e-1	2.5494e-1	2.5595e-1	**9.9976e-1**	9.9969e-1	**9.9976e-1**
PM1 (100)	2.8312e-1	8.1623e-2	8.2645e-3	5.7655e-2	2.1942e-1	2.1178e-1	1.5057e-1	9.5224e-2	3.3445e-1	3.0572e-1	**3.5000e-1**
PM2 (500)	1.7204e-1	1.3025e-1	8.2645e-3	9.0380e-2	1.2756e-1	1.2305e-1	1.4990e-1	9.7368e-2	**1.9429e-1**	1.6867e-1	1.8008e-1
PM3 (1000)	1.2228e-1	1.0926e-1	8.2645e-3	7.8952e-2	1.0825e-1	1.0755e-1	4.7697e-2	4.5595e-2	1.4634e-1	1.3407e-1	**2.0506e-1**
PM4 (5000)	8.7578e-2	9.5478e-2	8.2645e-3	6.6640e-2	9.6512e-2	9.5333e-2	1.2200e-1	6.2657e-2	1.1165e-1	1.0248e-1	**1.8328e-1**
CD1 (105)	6.6456e-1	5.5241e-1	4.2328e-1	5.4283e-1	6.8954e-1	6.7560e-1	6.4754e-1	5.1358e-1	6.2768e-1	5.9453e-1	**7.2724e-1**
CD2 (453)	7.1968e-1	5.5582e-1	4.0857e-1	5.7754e-1	7.4032e-1	7.4826e-1	6.8097e-1	5.0673e-1	5.3080e-1	6.0493e-1	**7.5426e-1**
CD3 (1133)	6.6584e-1	4.7085e-1	3.4262e-1	4.8436e-1	7.1771e-1	6.5668e-1	6.0266e-1	4.1732e-1	4.4462e-1	6.2029e-1	**7.5505e-1**
CD4 (4039)	6.1449e-1	4.5724e-1	3.2446e-1	4.4330e-1	6.9845e-1	6.5275e-1	5.9581e-1	3.9536e-1	4.0449e-1	5.7544e-1	**7.6296e-1**
CN1 (102)	9.0513e-1	7.7200e-1	6.3961e-1	7.5409e-1	9.0973e-1	8.7807e-1	8.7795e-1	7.5954e-1	9.2518e-1	9.2745e-1	**9.2792e-1**
CN2 (500)	8.6220e-1	7.2559e-1	5.9812e-1	7.2889e-1	9.7602e-1	9.5645e-1	9.2119e-1	7.1210e-1	9.8208e-1	**9.8335e-1**	9.5224e-1
CN3 (1000)	8.4360e-1	7.0641e-1	5.7786e-1	7.1164e-1	9.9777e-1	9.9625e-1	9.5660e-1	7.4710e-1	9.9833e-1	9.9818e-1	**9.9860e-1**
CN4 (5000)	8.3380e-1	7.0253e-1	5.6215e-1	6.8867e-1	9.9900e-1	9.9844e-1	9.6591e-1	7.1954e-1	9.9937e-1	9.9939e-1	**9.9946e-1**
PO1 (100)	9.8922e-2	1.0477e-1	9.3647e-2	9.6665e-2	9.9460e-2	9.6591e-2	1.1899e-1	1.1665e-1	1.2382e-1	1.2433e-1	**1.2458e-1**
PO2 (500)	9.3142e-2	9.5953e-2	9.1493e-2	9.2746e-2	9.3684e-2	9.2428e-2	1.1460e-1	1.0973e-1	1.2374e-1	**1.2464e-1**	1.1747e-1
PO3 (1000)	9.2252e-2	9.4490e-2	9.1168e-2	9.2043e-2	9.2646e-2	9.1741e-2	1.1031e-1	1.0622e-1	1.2378e-1	1.2413e-1	**1.2456e-1**
PO4 (5000)	9.1318e-2	9.1970e-2	9.0988e-2	9.1280e-2	9.1418e-2	9.1259e-2	1.0221e-1	9.7699e-2	**1.2494e-1**	1.2457e-1	1.2387e-1
KP1 (100)	7.4078e-2	0.0000e+0	0.0000e+0	0.0000e+0	4.7146e-2	5.5309e-2	6.3722e-3	4.3875e-2	8.8037e-2	**9.0874e-2**	8.8215e-2
KP2 (500)	8.2443e-2	0.0000e+0	3.8694e-2	0.0000e+0	6.2569e-2	6.0454e-2	0.0000e+0	4.6692e-2	8.1820e-2	**8.2996e-2**	7.8282e-2
KP3 (1000)	7.1668e-2	0.0000e+0	3.6040e-2	0.0000e+0	5.6379e-2	3.1971e-2	0.0000e+0	4.1131e-2	7.6705e-2	**7.9945e-2**	7.6881e-2
KP4 (5000)	8.1896e-2	0.0000e+0	5.1387e-2	0.0000e+0	6.2676e-2	3.6947e-2	0.0000e+0	4.6799e-2	7.8732e-2	**8.4221e-2**	8.1078e-2
Average ranking	5.3571	7.5714	10.3929	8.1429	5.0357	6.1786	6.3214	7.8929	3.1429	2.6429	1.8571

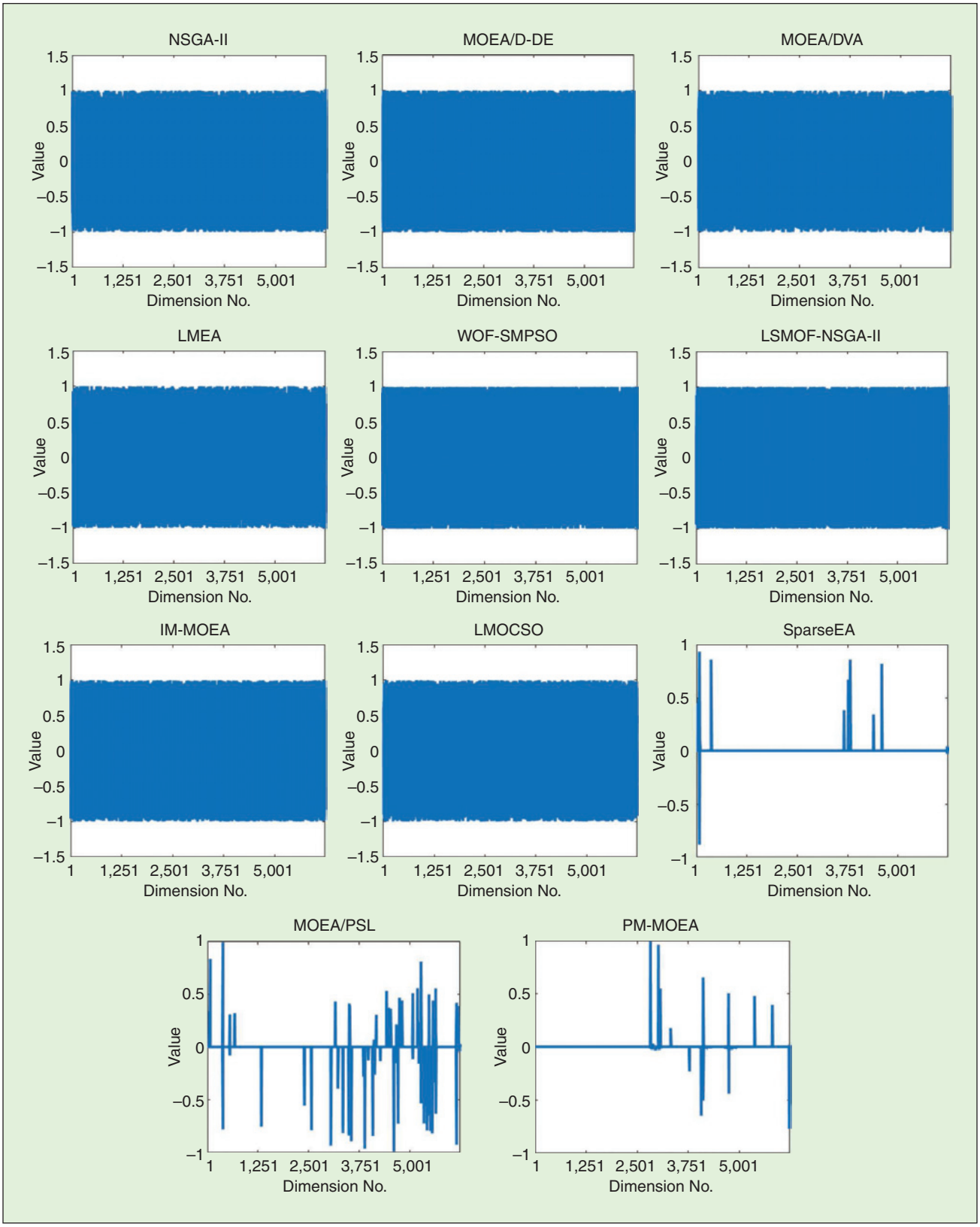


FIGURE 6 Parallel coordinates of solution sets (in decision space) with median CSD obtained by 11 MOEAs on neural network training with 6241 decision variables.

different from benchmark problems, including highly discrete decision spaces and irregular Pareto fronts. In particular, the objective functions of feature selection, pattern mining, community detection, critical node detection, and knapsack problem are calculated based on binary variables, which pose challenges to many MOEAs (i.e., MOEA/D-DE, MOEA/DVA, LMEA, WOF-SMPSO, LSMOF-NSGA-II, IM-MOEA, and LMOCSO) that can only handle real variables. Although these MOEAs can optimize real variables and round them before function evaluations, the introduction of flat landscapes is likely to deteriorate their search abilities. As evidenced by the CSD values shown in Table V, these MOEAs are underperformed by the binary genetic operator based NSGA-II. Although the uniform crossover and bit-flip mutation are the simplest genetic operators, they are relatively effective for solving binary MOPs and bring passable performance to NSGA-II.

Nevertheless, the high-dimensional decision spaces hinder NSGA-II from gaining the best CSD values on most test instances. By contrast, the three MOEAs tailored for sparse MOPs still exhibit the best overall performance, where SparseEA and MOEA/PSL obtain the best average ranking and are followed by PM-MOEA. Fig. 5 plots the objective values of the solution sets with median CSD obtained by the compared MOEAs on neural network training (NN4), portfolio optimization (PO4), and community detection (CD4) with approximately 5000 decision variables. For NN4, it can be found that MOEA/D-DE, MOEA/DVA, and LMOCSO cannot obtain good solutions in terms of the first objective (i.e., model complexity) and the second objective (i.e., error rate), whereas NSGA-II, LMEA, WOF-SMPSO, LSMOF-NSGA-II, and IM-MOEA can obtain a few solutions with extremely low error rates, but these solutions (i.e., neural networks) probably overfit the training set as analyzed in [20]. By contrast, SparseEA, MOEA/PSL, and PM-MOEA can obtain many solutions with both low complexities and error rates, where the low complexities can effectively alleviate overfitting. For PO4, only the solutions obtained by SparseEA, MOEA/PSL, and PM-MOEA hold good spread along the Pareto front, while the solutions obtained by all the other MOEAs shrink to the upper left corner. For CD4, the solution set obtained by PM-MOEA is well converged and diversified, the solution sets obtained by NSGA-II, SparseEA, and MOEA/PSL have good convergence but bad spread, and the solution sets obtained by all the other MOEAs are badly converged. As a result, the superiority of the sparse MOEAs is verified on both benchmark problems and real-world applications.

It is worth noting that SparseEA has worse average ranking than MOEA/PSL and PM-MOEA on benchmark problems but competitive average ranking on real-world applications. To illustrate this inconsistency, Table VI shows the HV values obtained by the compared MOEAs, where the reference point $(1, 1, \ldots, 1)$ is used for HV calculation. It can be found from the table that SparseEA obtains worse HV values than MOEA/PSL and PM-MOEA, which is inconsistent with the results in Table V. In particular, SparseEA has better CSD value but worse HV value than PM-MOEA on NN4. According to the solution sets obtained for NN4 shown in Fig. 5, the solutions obtained by SparseEA have slightly worse convergence than those obtained by PM-MOEA, leading to slightly worse HV value of SparseEA. On the other hand, Fig. 6 draws the decision variables of the solution sets obtained for NN4, where SparseEA, MOEA/PSL, and PM-MOEA obtain very sparse solution sets and the solution set obtained by SparseEA is the sparsest, hence SparseEA obtains the best CSD value on NN4. To summarize, the solution sets obtained by SparseEA have slightly worse convergence but better sparsity than those obtained by PM-MOEA, hence SparseEA gains better CSD values but worse HV values than PM-MOEA. Moreover, the effectiveness of the proposed indicator in the performance assessment of convergence, sparsity, and diversity can be further demonstrated.

D. Further Verification of the Proposed CSD

To further verify the effectiveness of the proposed CSD in separately assessing convergence, diversity, and sparsity, several

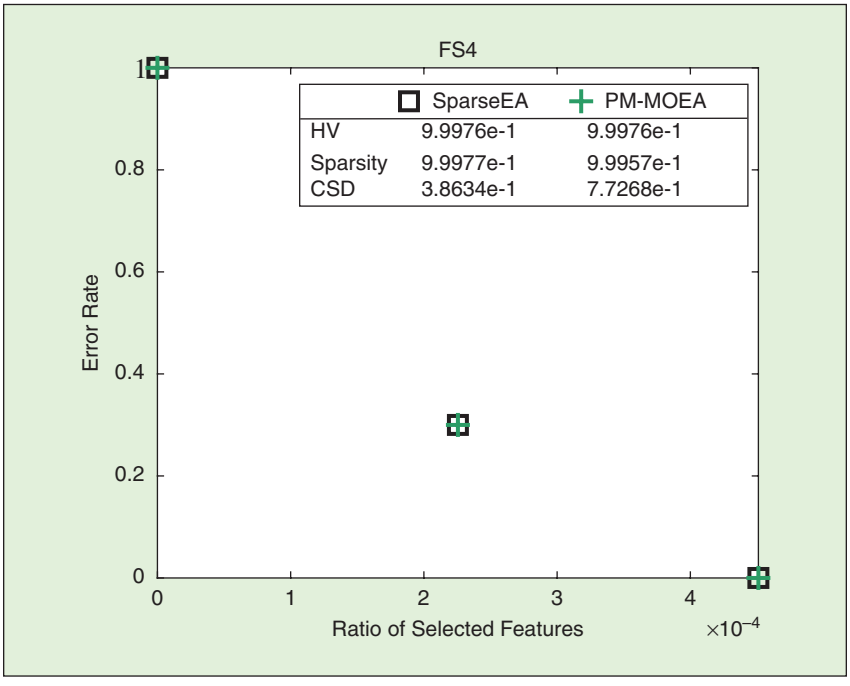


FIGURE 7 Two solution sets (in objective space) obtained by SparseEA and PM-MOEA on FS4, which have the same convergence and diversity but different sparsity.

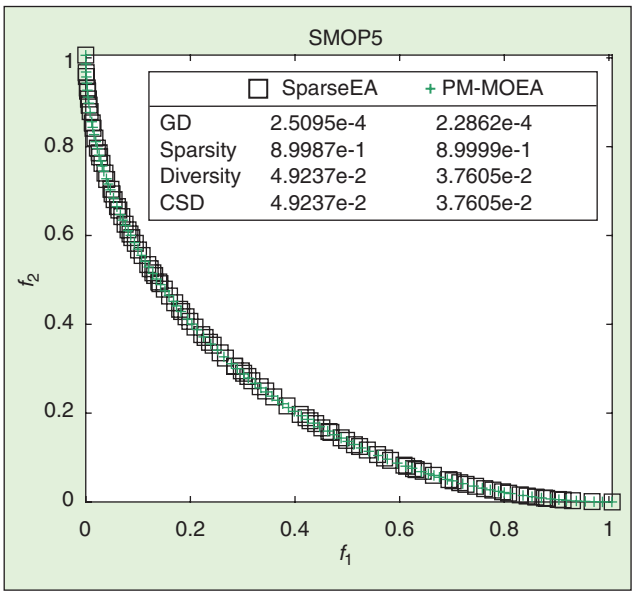


FIGURE 8 Two solution sets (in objective space) obtained by SparseEA and PM-MOEA on SMOP5, which have the same convergence and sparsity but different diversity.

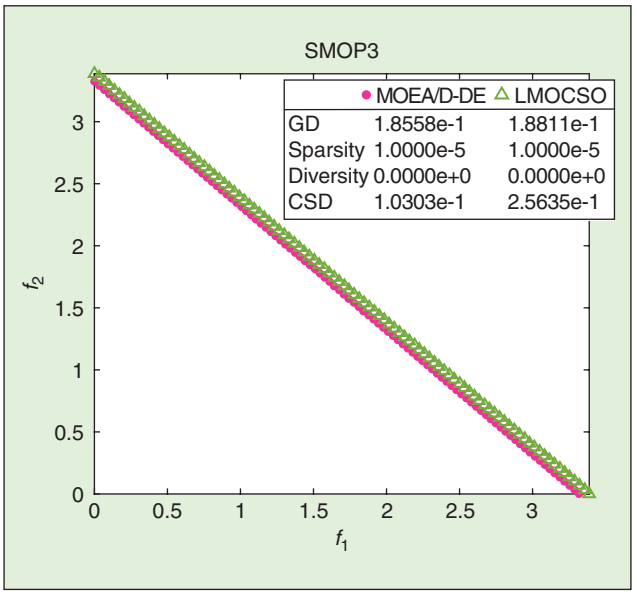


FIGURE 9 Two solution sets (in objective space) obtained by MOEA/D-DE and LMOCSO on SMOP3, which have the same diversity and sparsity but different convergence.

typical solution sets obtained by the compared MOEAs are investigated. Firstly, Fig. 7 plots two solution sets obtained by SparseEA and PM-MOEA on FS4, which are completely the same and thus have the same HV value. However, the two solution sets have different sparsity, thus leading to distinct CSD values. Secondly, Fig. 8 draws two solution sets obtained by SparseEA and PM-MOEA on SMOP5, which have very similar convergence (i.e., GD values) and sparsity. However, the solution set obtained by PM-MOEA distributes more uniformly than that obtained by SparseEA, thus leading to different diversity as well as CSD values. Lastly, Fig. 9 depicts two solution sets obtained by MOEA/D-DE and LMOCSO on SMOP3, which have the same sparsity and diversity. However, the solution set obtained by MOEA/D-DE converges better than that obtained by LMOCSO, thus leading to different GD values as well as CSD values. As a consequence, the proposed indicator is capable of distinguishing between the solution sets with only different convergence, diversity, or sparsity.

V. Conclusions

The sparsity of solutions is a pivotal factor in decision making, but it is ignored by most MOEAs and performance indicators. Therefore, this work has proposed a comprehensive indicator to compare the performance of MOEAs on large-scale sparse MOPs. While existing indicators use predefined reference points to assess only the convergence and diversity of the solution sets obtained by MOEAs, the proposed indicator can assess the convergence, diversity, and sparsity without the assistance of any reference point. According to the experiments, the proposed indicator is more suitable for assessing the quality of the solution sets for sparse MOPs than existing indicators.

The experimental results have indicated that large-scale MOEAs exhibit similar performance to classical MOEAs on large-scale sparse MOPs, even though the former suggested various search strategies to tackle high-dimensional decision spaces. By contrast, the sparse MOEAs proposed in recent years significantly outperform large-scale MOEAs and classical MOEAs, which is mainly due to their effectiveness in maintaining the sparsity of solutions. However, the performance of these sparse MOEAs fluctuates on different problems, which implies that their performance can be further enhanced. Since the proposed indicator is designed for assessing the convergence, diversity, and sparsity of solution sets, it is reasonable to embed the proposed indicator in the selection strategies of MOEAs to improve their performance on large-scale sparse MOPs.

Acknowledgment

This work was supported in part by the National Key R&D Program of China under Grant 2018AAA0100100, in part by the National Natural Science Foundation of China under Grant 61822301, Grant 61876123, Grant 61876162, Grant 61906001, Grant 62136008, Grant 62172002, and Grant U21A20512, in part by the Collaborative Innovation Program of Universities in Anhui Province under Grant GXXT-2020-013 and Grant GXXT-2020-051, and in part by the Research Grants Council of the Hong Kong Special Administrative Region, China under Grant PolyU11202418, Grant PolyU11209219, and Grant PolyU11211521.

References

[1] A. Zhou, B. Y. Qu, H. Li, S. Z. Zhao, P. N. Suganthan, and Q. Zhang, "Multiobjective evolutionary algorithms: A survey of the state of the art," *Swarm Evolut. Comput.*, vol. 1, no. 1, pp. 32–49, 2011, doi: 10.1016/j.swevo.2011.03.001.

[2] L. M. Antonio and C. A. Coello Coello, "Use of cooperative coevolution for solving large scale multiobjective

optimization problems," in *Proc. 2013 IEEE Congr. Evol. Comput.*, pp. 2758–2765.
[3] M. N. Omidvar, X. Li, Y. Mei, and X. Yao, "Cooperative co-evolution with differential grouping for large scale optimization," *IEEE Trans. Evol. Comput.*, vol. 18, no. 3, pp. 378–393, Jun. 2014, doi: 10.1109/TEVC.2013.2281543.
[4] F. Sander, H. Zille, and S. Mostaghim, "Transfer strategies from single- to multi-objective grouping mechanisms," in *Proc. 2018 Annu. Conf. Genetic Evol. Comput. Conf.*, pp. 729–736.
[5] X. Ma *et al.*, "A multiobjective evolutionary algorithm based on decision variable analyses for multiobjective optimization problems with large-scale variables," *IEEE Trans. Evol. Comput.*, vol. 20, no. 2, pp. 275–298, Apr. 2016, doi: 10.1109/TEVC.2015.2455812.
[6] X. Zhang, Y. Tian, R. Cheng, and Y. Jin, "A decision variable clustering-based evolutionary algorithm for large-scale many-objective optimization," *IEEE Trans. Evol. Comput.*, vol. 22, no. 1, pp. 97–112, Feb. 2018, doi: 10.1109/TEVC.2016.2600642.
[7] H. Zille, H. Ishibuchi, S. Mostaghim, and Y. Nojima, "A framework for large-scale multiobjective optimization based on problem transformation," *IEEE Trans. Evol. Comput.*, vol. 22, no. 2, pp. 260–275, Apr. 2018, doi: 10.1109/TEVC.2017.2704782.
[8] C. He *et al.*, "Accelerating large-scale multiobjective optimization via problem reformulation," *IEEE Trans. Evol. Comput.*, vol. 23, no. 6, pp. 949–961, Dec. 2019, doi: 10.1109/TEVC.2019.2896002.
[9] H. Qian and Y. Yu, "Solving high-dimensional multi-objective optimization problems with low effective dimensions," in *Proc. 31st AAAI Conf. Artif. Intell.*, 2017, pp. 875–881.
[10] R. Liu, R. Ren, J. Liu, and J. Liu, "A clustering and dimensionality reduction based evolutionary algorithm for large-scale multi-objective problems," *Appl. Soft Comput.*, vol. 89, p. 106,120, 2020.
[11] Y. Tian, X. Zheng, X. Zhang, and Y. Jin, "Efficient large-scale multi-objective optimization based on a competitive swarm optimizer," *IEEE Trans. Cybern.*, vol. 50, no. 8, pp. 3696–3708, Aug. 2020, doi: 10.1109/TCYB.2019.2906383.
[12] C. He, R. Cheng, and D. Yazdani, "Adaptive offspring generation for evolutionary large-scale multiobjective optimization," *IEEE Trans. Syst., Man, Cybern. Syst.*, vol. 52, no. 2, pp. 786–798 Feb 2022, doi: 10.1109/TSMC.2020.3003926.
[13] W. Hong, K. Tang, A. Zhou, H. Ishibuchi, and X. Yao, "A scalable indicator-based evolutionary algorithm for large-scale multiobjective optimization," *IEEE Trans. Evol. Comput.*, vol. 23, no. 3, pp. 525–537, Jun. 2018, doi: 10.1109/TEVC.2018.2881153.
[14] H. Chen, R. Cheng, J. Wen, H. Li, and J. Weng, "Solving large-scale many-objective optimization problems by covariance matrix adaptation evolution strategy with scalable small subpopulations," *Inf. Sci.*, vol. 509, pp. 457–469, 2020, doi: 10.1016/j.ins.2018.10.007.
[15] Y. Jin and B. Sendhoff, "Pareto-based multiobjective machine learning: An overview and case studies," *IEEE Trans. Syst., Man, Cybern. C, Appl. Rev.*, vol. 38, no. 3, pp. 397–415, May 2008, doi: 10.1109/TSMCC.2008.919172.
[16] Y. Tian, S. Yang, L. Zhang, F. Duan, and X. Zhang, "A surrogate-assisted multiobjective evolutionary algorithm for large-scale task-oriented pattern mining," *IEEE Trans. Emerg. Topics Comput. Intell.*, vol. 3, no. 2, pp. 106–116, Apr. 2019, doi: 10.1109/TETCI.2018.2872055.
[17] M. Lalou, M. A. Tahraoui, and H. Kheddouci, "The critical node detection problem in networks: A survey," *Comput. Sci. Rev.*, vol. 28, pp. 92–117, 2018, doi: 10.1016/j.cosrev.2018.02.002.
[18] H. Li, Q. Zhang, J. Deng, and B. X. Zong, "A preference-based multiobjective evolutionary approach for sparse optimization," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 29, no. 5, pp. 1716–1731, May 2018, doi: 10.1109/TNNLS.2017.2677973.
[19] A. Ponsich, A. L. Jaimes, and C. A. Coello Coello, "A survey on multiobjective evolutionary algorithms for the solution of the portfolio optimization problem and other finance and economics applications," *IEEE Trans. Evol. Comput.*, vol. 17, no. 3, pp. 321–344, Jun. 2013, doi: 10.1109/TEVC.2012.2196800.
[20] Y. Tian, X. Zhang, C. Wang, and Y. Jin, "An evolutionary algorithm for large-scale sparse multiobjective optimization problems," *IEEE Trans. Evol. Comput.*, vol. 24, no. 2, pp. 380–393, Apr. 2020, doi: 10.1109/TEVC.2019.2918140.
[21] Z. Tan, H. Wang, and S. Liu, "Multi-stage dimension reduction for expensive sparse multi-objective optimization problems," *Neurocomputing*, vol. 440, pp. 159–174, 2021, doi: 10.1016/j.neucom.2021.01.115.
[22] Y. Tian, C. Lu, X. Zhang, K. C. Tan, and Y. Jin, "Solving large-scale multi-objective optimization problems with sparse optimal solutions via unsupervised neural networks," *IEEE Trans. Cybern.*, vol. 51, no. 6, pp. 3115–3128, Jun. 2021, doi: 10.1109/TCYB.2020.2979930.
[23] C. A. C. Coello and N. C. Cortés, "Solving multiobjective optimization problems using an artificial immune system," *Genetic Program. Evolvable Mach.*, vol. 6, pp. 163–190, 2005, doi: 10.1007/s10710-005-6164-x.
[24] E. Zitzler and L. Thiele, "Multiobjective optimization using evolutionary algorithms-a comparative case study," in *Proc. 1998 Int. Conf. Parallel Problem Solving Nature*, pp. 292–301.
[25] H. Ishibuchi, R. Imada, Y. Setoguchi, and Y. Nojima, "Reference point specification in hypervolume calculation for fair comparison and efficient search," in *Proc. 2017 Genetic Evol. Comput. Conf.*, pp. 585–592, doi: 10.1145/3071178.3071264.
[26] Y. Tian, R. Cheng, X. Zhang, M. Li, and Y. Jin, "Diversity assessment of multi-objective evolutionary algorithms: Performance metric and benchmark problems [Research Frontier]," *IEEE Comput. Intell. Mag.*, vol. 14, no. 3, pp. 61–74, Aug. 2019, doi: 10.1109/MCI.2019.2919398.
[27] Y. Tian, C. Lu, X. Zhang, F. Cheng, and Y. Jin, "A pattern mining based evolutionary algorithm for large-scale sparse multi-objective optimization problems," *IEEE Trans. Cybern.*, early access, Dec. 30, 2020, doi: 10.1109/TCYB.2020.3041325.
[28] M. Li and J. Wei, "A cooperative co-evolutionary algorithm for large-scale multi-objective optimization problems," in *Proc. 2018 Annu. Conf. Genetic Evol. Comput. Conf.*, pp. 1716–1721.
[29] K. Deb, A. Pratap, S. Agarwal, and T. Meyarivan, "A fast and elitist multiobjective genetic algorithm: NSGA-II," *IEEE Trans. Evol. Comput.*, vol. 6, no. 2, pp. 182–197, Apr. 2002, doi: 10.1109/4235.996017.
[30] H. Li and Q. Zhang, "Multiobjective optimization problems with complicated Pareto sets, MOEA/D and NSGA-II," *IEEE Trans. Evol. Comput.*, vol. 13, no. 2, pp. 284–302, Apr. 2009, doi: 10.1109/TEVC.2008.925798.
[31] B. Xue, M. Zhang, and W. N. Browne, "Particle swarm optimization for feature selection in classification: A multi-objective approach," *IEEE Trans. Cybern.*, vol. 43, no. 6, pp. 1656–1671, Dec. 2013, doi: 10.1109/TSMCB.2012.2227469.
[32] R. Cheng, Y. Jin, K. Narukawa, and B. Sendhoff, "A multiobjective evolutionary algorithm using Gaussian process-based inverse modeling," *IEEE Trans. Evol. Comput.*, vol. 19, no. 6, pp. 838–856, Dec. 2015, doi: 10.1109/TEVC.2015.2395073.
[33] A. Fischer and C. Igel, "An introduction to restricted Boltzmann machines," in *Proc. Iberoam. Congr. Pattern Recognit.*, Springer-Verlag, 2012, pp. 14–36.
[34] P. Vincent, H. Larochelle, Y. Bengio, and P.-A. Manzagol, "Extracting and composing robust features with denoising autoencoders," in *Proc. 25th Int. Conf. Mach. Learn.*, 2008, pp. 1096–1103, doi: 10.1145/1390156.1390294.
[35] Y. Tian, R. Liu, X. Zhang, H. Ma, K. C. Tan, and Y. Jin, "A multi-population evolutionary algorithm for solving large-scale multi-modal multi-objective optimization problems," *IEEE Trans. Evol. Comput.*, vol. 25, no. 3, pp. 405–418, Jun. 2021, doi: 10.1109/TEVC.2020.3044711.
[36] M. Li and X. Yao, "Quality evaluation of solution sets in multiobjective optimisation," *ACM Comput. Surv.*, vol. 52, no. 2, pp. 1–38, 2019, doi: 10.1145/3300148.
[37] K. C. Tan, T. H. Lee, and E. F. Khor, "Evolutionary algorithms for multi-objective optimization: Performance assessments and comparisons," in *Proc. 2001 IEEE Congr. Evol. Comput.*, pp. 979–986.
[38] S. Bandyopadhyay, S. K. Pal, and B. Aruna, "Multiobjective GAs, quantitative indices, and pattern classification," *IEEE Trans. Syst., Man, Cybern. B, Cybern.*, vol. 34, no. 5, pp. 2088–2099, Oct. 2004, doi: 10.1109/TSMCB.2004.834438.
[39] D. Veldhuizen and G. Lamont, "Evolutionary computation and convergence to a Pareto front," in *Late Breaking Papers 1998 Genetic Program. Conf.*, 1998, pp. 221–228.
[40] J. R. Schott, "Fault tolerant design using single and multi-criteria genetic algorithms," M.S. thesis, Department of Aeronautics and Astronautics, Massachusetts Institute of Technology, 1995.
[41] J. Wu and S. Azarm, "Metrics for quality assessment of a multiobjective design optimization solution set," *J. Mech. Des.*, vol. 123, no. 1, pp. 18–25, 2001, doi: 10.1115/1.1329875.
[42] Y. Tian, X. Xiang, X. Zhang, R. Cheng, and Y. Jin, "Sampling reference points on the Pareto fronts of benchmark multi-objective optimization problems," in *Proc. 2018 IEEE Congr. Evol. Comput.*
[43] A. Zhou, Q. Zhang, and Y. Jin, "Approximating the set of Pareto-optimal solutions in both the decision and objective spaces by an estimation of distribution algorithm," *IEEE Trans. Evol. Comput.*, vol. 13, no. 5, pp. 1167–1189, Oct. 2009, doi: 10.1109/TEVC.2009.2021467.
[44] X. Zhang, Y. Tian, R. Cheng, and Y. Jin, "An efficient approach to nondominated sorting for evolutionary multiobjective optimization," *IEEE Trans. Evol. Comput.*, vol. 19, no. 2, pp. 201–213, Apr. 2015, doi: 10.1109/TEVC.2014.2308305.
[45] G. Lizárraga-Lizárraga, A. Hernández-Aguirre, and S. Botello-Rionda, "G-metric: An M-ary quality indicator for the evaluation of nondominated sets," in *Proc. 10th Annu. Conf. Genetic Evol. Comput.*, 2008, pp. 665–672.
[46] Q. Zhang and H. Li, "MOEA/D: A multiobjective evolutionary algorithm based on decomposition," *IEEE Trans. Evol. Comput.*, vol. 11, no. 6, pp. 712–731, Dec. 2007, doi: 10.1109/TEVC.2007.892759.
[47] X. Cai, H. Sun, and Z. Fan, "A diversity indicator based on reference vectors for many-objective optimization," *Inf. Sci.*, vols. 430-431, pp. 467–486, 2018, doi: 10.1016/j.ins.2017.11.051.
[48] Y. Tian, X. Xiang, X. Zhang, R. Cheng, and Y. Jin, "Sampling reference points on the Pareto fronts of benchmark multi-objective optimization problems," in *Proc. 2018 IEEE Congr. Evol. Comput.*
[49] M. Li, S. Yang, and X. Liu, "Diversity comparison of Pareto front approximations in many-objective optimization," *IEEE Trans. Cybern.*, vol. 44, no. 12, pp. 2568–2584, Dec. 2014, doi: 10.1109/TCYB.2014.2310651.
[50] R. Agrawal and R. Srikant, "Fast algorithms for mining association rules," in *Proc. Int. Conf. Very Large Data Bases*, 1994, pp. 487–499.
[51] R. A. Rossi and N. K. Ahmed, "The network data repository with interactive graph analytics and visualization," in *Proc. 29th AAAI Conf. Artif. Intell.*, 2015. [Online]. Available: http://networkrepository.com
[52] E. Zitzler and L. Thiele, "Multiobjective evolutionary algorithms: A comparative case study and the strength pareto approach," *IEEE Trans. Evol. Comput.*, vol. 3, no. 4, pp. 257–271, Nov. 1999, doi: 10.1109/4235.797969.
[53] Y. Tian, R. Cheng, X. Zhang, and Y. Jin, "PlatEMO: A MATLAB platform for evolutionary multi-objective optimization [Educational Forum]," *IEEE Comput. Intell. Mag.*, vol. 12, no. 4, pp. 73–87, Nov. 2017, doi: 10.1109/MCI.2017.2742868.
[54] R. B. Agrawal, K. Deb, and R. B. Agrawal, "Simulated binary crossover for continuous search space," *Complex Syst.*, vol. 9, no. 4, pp. 115–148, 1995.
[55] K. Deb and M. Goyal, "A combined genetic adaptive search (GeneAS) for engineering design," *Comput. Sci. Informat.*, vol. 26, no. 4, pp. 33–45, 1996.
[56] J. Derrac, S. Garcia, D. Molina, and F. Herrera, "A practical tutorial on the use of nonparametric statistical tests as a methodology for comparing evolutionary and swarm intelligence algorithms," *Swarm Evol. Comput.*, vol. 1, no. 1, pp. 3–18, 2011, doi: 10.1016/j.swevo.2011.02.002.

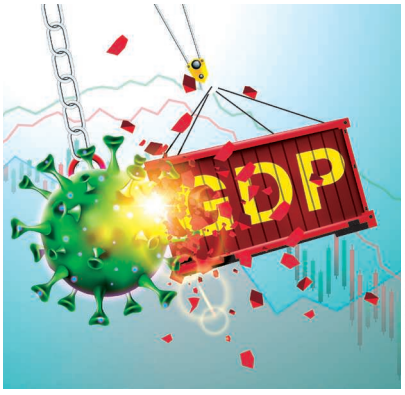
Dominik Fischer,
Sanaz Mostaghim, and
Thomas Seidelmann
Otto von Guericke University
Magdeburg, GERMANY

Exploring Dynamic Pandemic Containment Strategies Using Multi-Objective Optimization

Abstract

The SARS-CoV-2 pandemic demonstrates the vulnerability of societies in a globalized world. As pathogens spread at exponential rates, rapid development of appropriate medical treatments and distribution of vaccinations are major challenges. Under these circumstances, authorities employ non-pharmaceutical interventions (NPIs) against the spread, which can impact the economy strongly. Hence, there is a need for strategies that help minimize infection without sacrificing the economy's wellbeing. This study explores the inherent trade-off characteristics of optimal control strategies by utilizing the well-known SEIR (susceptible, exposed, infectious, recovered) pandemic model with an integrated economic compartment. The health economy dilemma (HED) qualifies as a multi-objective optimization (MOO) problem and the goal is to find strategies which are optimal regarding concurrent infections, economic growth, and required intensity of employed interventions. The major contribution of this paper is to propose a new methodology for containment strategy exploration using MOO. The experiments show that the resulting solutions can contribute towards solving the HED by supporting the identification of optimal strategies. Specific characteristics of pandemics are highlighted in this novel tri-objective optimization approach.

Digital Object Identifier 10.1109/MCI.2022.3181347
Date of current version: 19 July 2022



I. Introduction

The SARS-CoV-2 pandemic highlights the importance of effective non-pharmaceutical interventions (NPIs) to reduce the pathogen's spreading rate [1] and thus prevent a health-care system collapse and peak fatalities [2]. Since NPIs restrict individuals' behavior, they have a large impact on both economic and social systems. It is a major challenge to find an appropriate strategy that simultaneously considers health, economy and social aspects, which are known to be in conflict with each other. For instance, the best strategy regarding infection numbers in such scenarios is to quarantine the population, but this is not optimal for the economy. Such problems are usually known as multi-objective optimization problems (MOPs). There is a lack of effective tools for identifying optimal control strategies that respect the multi-objective decision-making process [3]. Machine learning and evolutionary algorithms offer methods to develop such tools, among which are single objective approaches that minimize infection numbers and bi-objective approaches that include economic costs [4]–[6].

Most approaches use predefined containment strategies or static estimations of those strategies. Generally, only time and duration of the interventions are optimized [3], [4]. Common optimization models concentrate on few pandemic parameters, such as manipulation of the contact rate, while neglecting other parameters that influence the diffusion process. The goal of this paper is to explore novel containment strategies using multi-objective optimization. This study extends the work in [7], which integrates the pathogen diffusion with a predictor for relative economic health and enables the dynamic application of control policies. However, they studied only two strategies, namely social distancing and lockdown, which are fixed in duration and strength. Here, Salgotra et al.'s approach is improved by supporting a wider range of policies and by enhancing the flexibility of their usage. In addition, the SEIR model [8], [9] is extended, too.

Another contribution of this paper is to study the above problem as a tri-objective optimization problem: Minimization of infection peaks, minimization of economic damage (as in [7]), and minimization of the cost of containment measures. The problem is solved by four state-of-the-art multi-objective optimization algorithms and their performance is compared. The algorithms find strategies that are sensitive to the current state of the pandemic and are effective even at moderate strength, although higher

intervention efforts are required to reconcile health and economy optimally. A hypothetical pandemic scenario, similar to the one presented in [7], is used for the experiments, which allows the comparison of results between these papers.

II. Background and Related Work

NPIs try to minimize the exposure to the pathogen, with the goal to reduce the infection risk. Increased hygiene, social distancing, lockdowns, and contact tracing are examples for typical NPIs [1], [10]. Despite NPIs being very effective at containing pandemics, they impose limitations on the economic system, e.g., social-distancing rules burden the food-service industry, and border closures shut down large parts of the travel industry. Hence, decision-makers face the challenge of choosing containment strategies that prevent the pathogen from spreading without burdening the economy more than necessary.

This can be modeled by the health economy dilemma (HED), which can be formulated as a multi-objective optimization problem. In the literature, several mathematical diffusion models are used either to estimate the model parameters for a real-world fit, or to find better containment strategies. Various works focus on optimization or prediction of the pandemic's variables [11], while a minority of the works model decision-making for containment strategies as a multi-objective problem (e.g., [6], [7], [12], [13]). For their optimization, control strategies have to be represented within the diffusion models with adjustable parameters. Usually, the interventions are picked using secondary data [6], determined by machine learning [4], or estimated using real-world proxy measures [14]. Either way, the resulting strategies are based on interventions already known. Hence, novel strategies are unlikely to be found, a challenge that this paper addresses.

Depending on the level of analysis, various diffusion models have been studied in the literature. On a fine-grained level, agent-based models, which respect stochastic interactions between individuals, are common [15], [16]. These models integrate at least one random variable, such as the transmission probability. Using these models to study large populations is a big challenge, as the data availability for large-scale applications is poor, which diminishes the advantages of the approach. Long-lasting pandemics exhibit dynamic behavior that is difficult to capture in a fine-grained model without compromising the accuracy of predictions [16]. In large-scale studies, decision-makers usually rely on compartment models, e.g., deterministic differential equations like variants of the SIR compartment model [2], [8]. The SIR (susceptible, infectious, recovered) model contains several compartments into which the population is divided. Similarly, the SEIR variant (susceptible, exposed, infectious, recovered) adds the exposed compartment that contains individuals who came into contact with a pathogen, but are not yet infected. Due to the simplicity of this model, it is often used for real-world simulations [7], [17].

It is a major challenge to find an appropriate strategy that simultaneously considers health, economy and social aspects, which are known to be in conflict with each other.

III. Methodology

The presented approach for the pandemic compartment model is fundamentally close to the common SEIR model [9]. The main compartments are: S (susceptible) for individuals who are vulnerable to being infected. E (exposed) for those who came into contact with infected individuals and will become infected. I (infected) for the spreaders of the virus. R (recovered) indicates those who have gained immunity. Typically, the population starts in S and then transitions over to E, I, and eventually R. The relative compartment sizes and equation parameters, such as the transmission probability, determine how fast this process happens and how the curves for each compartment will look like. Often, individuals in R will lose immunity over time and go back to S, in which case the virus will never fully disappear in these models. In addition, compartments for quarantined individuals were added analogous to the basic compartments: Sq, Eq, Iq, and Rq respectively. It is an assumption in this study that measures leading to quarantine of individuals are the most significant problem for the economy. Therefore, the inclusion of these specialized compartments allows modeling of this property. Additionally, the SEIR model is extended to include a compartment D (deaths) for individuals who succumbed to the virus. The economy is represented by a virtual compartment (see Section III-A). Figure 1 illustrates the links between compartments and highlights the most important driving forces between them. The population in each compartment is relative to the total population. As such, the total population is denoted by 1 (representing 100% of the population) and is equal to the sum of all pandemic compartments, with each being ≥ 0 and ≤ 1:

$$N = 1 = S + S_q + E + E_q + I + I_q + R + R_q + D \tag{1}$$

The differential equations which link the pandemic compartments are given by the following:

$$\frac{dS}{dt} = -c_r(1-t_p)c_{dp}IS - c_r t_p(1-c_{dp})IS - c_r t_p c_{dp} IS - p_{qr}S - v_r S + p_{qer}S_q + i_{lr}R \tag{2}$$

$$\frac{dS_q}{dt} = c_r(1-t_p)c_{dp}IS + p_{qr}S + i_{lr}R_q - p_{qer}S_q \tag{3}$$

$$\frac{dE}{dt} = c_r t_p(1-c_{dp})IS + p_{qer}E_q - p_{qr}E - i_r E \tag{4}$$

$$\frac{dE_q}{dt} = +c_r t_p c_{dp} IS + p_{qr}E - p_{qer}E_q - i_r E_q \tag{5}$$

$$\frac{dI}{dt} = +i_r E - d_r I - i_{rr} d_p I - i_{rr}(1-d_p) I \tag{6}$$

$$\frac{dI_q}{dt} = +i_r E_q + d_r I - i_{qrr} d_p I_q - i_{qrr}(1-d_p) I_q \tag{7}$$

$$\frac{dR}{dt} = +i_{rr}(1-d_p) I + i_{qrr}(1-d_p) I_q + \nu_r S + p_{qer} R_q - i_{lr} R - c_r c_{dp} IR - p_{qr} R \tag{8}$$

$$\frac{dR_q}{dt} = +c_r c_{dp} IR + p_{qr} R - p_{qer} R_q - i_{lr} R_q \tag{9}$$

$$\frac{dD}{dt} = +i_{rr} d_p I + i_{qrr} d_p I_q \tag{10}$$

The Equations (2) to (10) contain the following parameters: c_r indicates the contact rate. The transmission probability is shown by t_p. The incubation rate (i_r) specifies how long it takes from exposure to becoming infectious. The preventive quarantine rate (p_{qr}) determines the rate with which susceptible individuals are quarantined preventively, and the preventive quarantine end rate (p_{qer}) determines how long this confinement lasts. The contact detection probability (c_{dp}) defines the chance that the contact of a susceptible individual with an infected individuals is successfully traced, and that the susceptible individuals can be sent to quarantine based on this information. The diagnosis rate (d_r) describes the rate with which infected individuals are detected and isolated. The infected individuals' recovery rates are given by the infected recovery rate (i_{rr}) and infected quarantined recovery rate (i_{qrr}). The immunity loss rate (i_{lr}) determines how long-lasting immunity from the virus is. The vaccination rate (ν_r) determines how frequently susceptible individuals transition to immunity without requiring infection. Finally, the death probability is indicated by d_p. The presented pandemic model integrates control policies, by allowing the policies to alter these parameters during simulation (see Section III-B).

The equations are built from individual terms, where each term describes the flow of individuals from one compartment to another. Every term appears exactly twice, once as an addition to a compartment, once as a subtraction, thus guaranteeing that the total population remains constant. The first group of terms is concerned with the contact between susceptible, recovered and infected: $c_r(1-t_p) c_{dp} IS$ represents susceptible individuals who had a traced contact with someone who was infected, without becoming exposed. They are sent into preventive quarantine (S to S_q). Similarly, $c_r t_p c_{dp} IS$ represents susceptible individuals with traced contacts that were exposed and are sent into quarantine (S to E_q). The term $c_r t_p (1-c_{dp}) IS$ represents exposure, but without being detected (S to E). Individuals in the E compartment do not know of their exposure and therefore do not adapt their behavior. For instance, they still go to work and eat at restaurants. Next, $p_{qr} S$ are susceptible individuals ordered to stay in quarantine for prevention purposes (S to S_q), and $p_{qer} S_q$ represents individuals returning from such quarantine orders (S_q to S). It is assumed that even recovered individuals must obey quarantine rules, despite their immunity. Therefore, there are analogous terms for individuals entering and leaving preventive quarantine for the exposed and recovered compartment pairs. The next term, $c_r c_{dp} IR$, describes successfully traced contacts between infected individuals and already recovered individuals, leading to quarantine afterwards (R to R_q). The term $\nu_r S$ captures the vaccination of susceptible individuals (S to R), while $i_{lr} R$ and $i_{lr} R_q$ capture the loss of immunity (R to S, respectively R_q to S_q). After the incubation time, the exposed individuals show symptoms and transition to I: $i_r E$ (E to I). The same occurs for the quarantined versions of the compartments: $i_r E_q$ (E_q to I_q). A fraction of individuals that carry an infection get tested and receive a positive diagnosis, $d_r I$, which consequently puts them in quarantine (I to I_q). Finally, individuals either succumb to the pathogen, or recover and become immune. This applies to both I and I_q. Therefore, $i_{rr} d_p I$ and $i_{qrr} d_p I_q$ describe the rate of individuals succumbing (I/I_q to D), while $i_{rr}(1-d_p) I$ and $i_{qrr}(1-d_p) I_q$ represent the fraction of individuals that recover from the virus and become immune (I/I_q to R).

A. The Economy Compartment

The relative economic predictor of the proposed model is introduced via a virtual compartment, which is unidirectionally influenced by the pandemic compartments (see Figure 1). Hence, it is assumed that the pandemic's course shapes the economy, but not vice versa. This virtual compartment is called *GDP* (Gross Domestic Product). It is to be understood as a rough indicator for the state of the economy relative to the starting point of the pandemic:

$$\frac{dGDP}{dt} = +b_g(1-D) - p_i p_{qi} S_q - p_i e_{qi} E_q - p_i i_i I - p_i i_{qi} I_q - p_i p_{qi} R_q \tag{11}$$

The GDP is modeled by the characteristic of economic growth b_g of capitalist economies. Usually, an economy's GDP consists of the *Consumption, Investment, Government Spending,* and *Net Exports*—in this approach an overly complex representation of the economy. In the presented model, the GDP fundamentally grows linearly at a constant rate, although it scales with the population in this model (economic growth weakens as deaths accumulate). The remaining parameters quantify the influence of negatively associated compartments on economic health: S_q, E_q, I, I_q, and R_q. For each of these compartments, which counteract the fundamental economic growth, there is a scaling factor that describes the strength of its negative impact on the economy: preventive quarantine impact (p_{qi}), exposed quarantine impact (e_{qi}), infected impact (i_i), and infected quarantine impact (i_{qi}). Such scaling factors are required to represent different economies. For instance, high-tech economies are less impacted by a pandemic than those economies strongly built upon tourism. In addition, the model includes a general pandemic influence parameter (p_i) that serves as a simple scaling factor for the previously mentioned individual impacts. This way, overall impact can be adjusted easily without altering relative impacts, which proved useful in preliminary experiments.

B. Control Policies

In this study, the term *containment strategy* refers to a set of control policies. Each policy influences one specific parameter, has a trigger time ≥ 0 and $\leq t_n$, and is further described by a *Bézier curve* (see Figure 2 for an example). The curve represents the time in relation to strength of adjustment. The adjustments are additive, meaning that the value of each parameter at a given time t is equal to its base value plus the value of the policy for t. The curves are defined via 16 control points. The height (y-value) for each control point is ≥ 0 and ≤ 1 and represents the relative policy strength. The points #1 and #2 are always fixed at (0, 0) and (10, 0), because it is assumed that policies must start from zero strength at the beginning of the pandemic. While the next 14 points also have fixed times, their heights are considered variable and are subject to optimization in the experiments to find optimal values for them. The point #3 has time 15, and a height always equal to the point #4, with a time of 20. The idea is to group these unusually close together to facilitate a rapid response from zero strength to the onset of the pandemic. All following points are regularly spaced, with independent height values. The last point has a time of t_n. The curve is still constructed exactly as described above for policies with a trigger time > 0. However, the curve is subsequently scaled on the x-axis so that the x-value of the first point coincides with the trigger time, while the last point remains at t_n.

C. Optimization

Given the above model, the multi-objective optimization problem can be defined as follows. The decision space of the problem is built by the parameters related to the control policies. The policies have fixed trigger times, and the first three control points for each policy are predefined. That leaves 13 control points per policy for which the simulation experiments need to find optimal y-values. All other model parameters, including the number and type of policies to be optimized, are predefined and constant throughout the optimization. The vector of decision variables is denoted as d and the vector of all other predefined parameters as m. Therefore, each individual solution s is defined by the individual vectors d and m. Applying

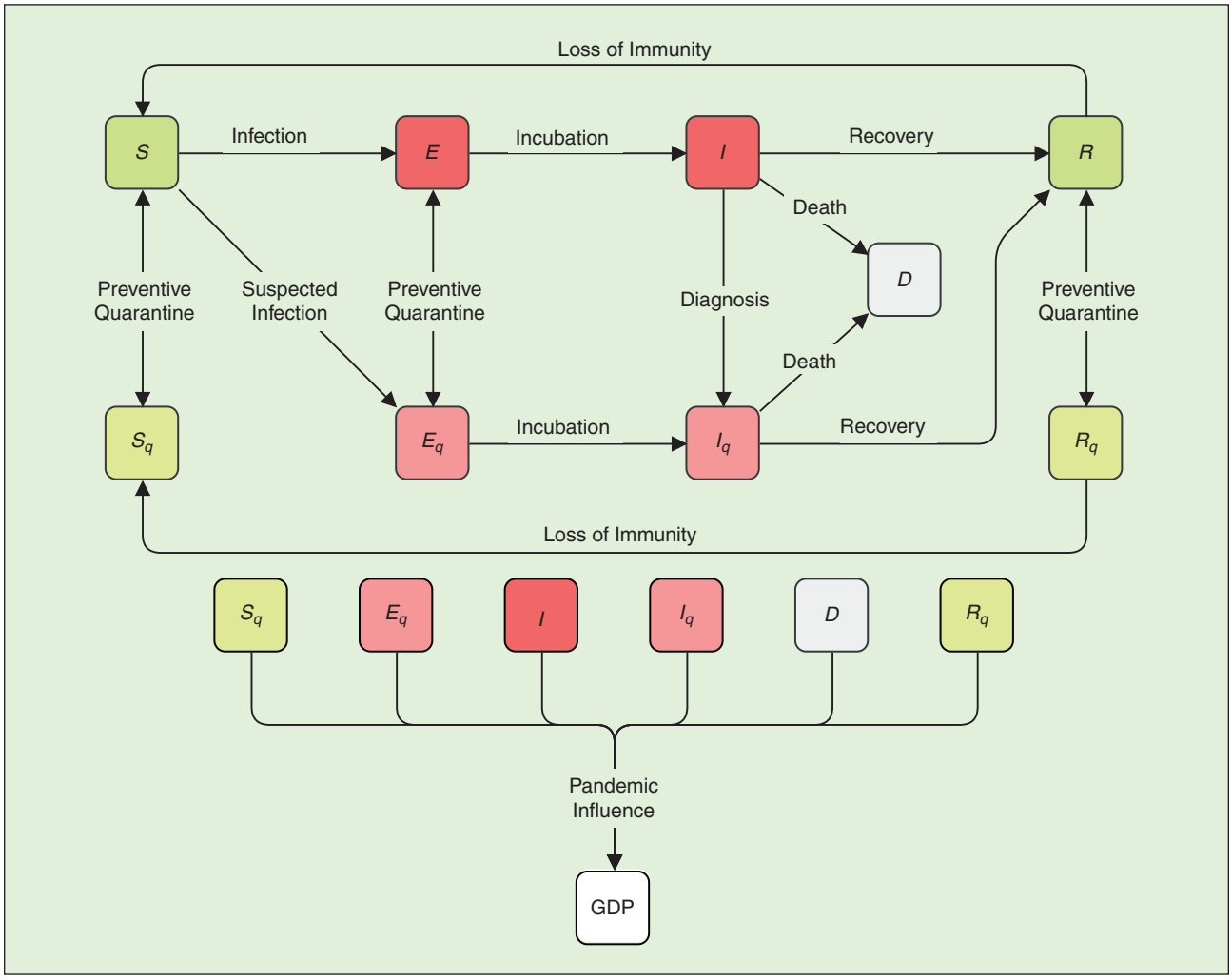


FIGURE 1 The top graph shows the model compartments for the pandemic simulation. The bottom graph shows which of the pandemic compartments are influencing the economy compartment.

the simulation model to s returns a set of vectors with compartment values for each time step $1, \ldots, t_n$. Therefore, $s(S)$ refers to the resulting vector of S compartment values, $s(E)$ to the vector of E compartment values, etc. Three objective functions (f_1, f_2, f_3) assess the performance of the solutions and their values should be minimized by the optimization algorithms. The first two are based on [7], and the third is a new addition.

a) Health Objective

The first objective (f_1) corresponds to the peak of concurrent infections.

$$f_1(s) = \max\left(s(E) + s(E_q) + s(I) + s(I_q)\right) \quad (12)$$

It can be assumed that minimization of this function is most important for preventing a collapse of the health care system. Using the total number of infections over the simulated period does not necessarily penalize high infection peaks, therefore allowing overloads of the health care systems.

b) Economy Objective

The second objective (f_2) represents the goal of minimizing economic setbacks.

$$f_2(s) = -s(GDP)_{t_n} \quad (13)$$

In the reference work [7] this objective is modelled by measuring the peak damage to GDP. Since the GDP is defined to start at 0 in this model, the objective value cannot become better than that. Any solutions which keep the GDP curve at or above 0 for the entire duration would be equal and optimal for the algorithm. The approach used in this paper defines the objective as the negative value of the *GDP* at the last time step (t_n), which enables the distinction between solutions that successfully prevent an economic setback compared to the starting value.

c) Intervention Efforts Objective

The third objective (f_3) measures the overall strength of the employed containment strategies. Let P be the set of policy strength curves for all policies in s, with height ≥ 0 and ≤ 1. Furthermore, let $\bar{p}$ for each $p \in P$ be the average height of the curve. Then f_3 is given by the following.

$$f_3(s) = \frac{\sum_{p \in P} \bar{p}}{|P|} \quad (14)$$

In this study, less regulatory influence is considered to be desirable for a healthy society. As this objective is in direct conflict with the usage of strong policies, its purpose is to force the optimization process to find efficient and effective solutions simultaneously. This objective is calculated by taking the average normalized influence of all parameter adjustment curves for an experiment.

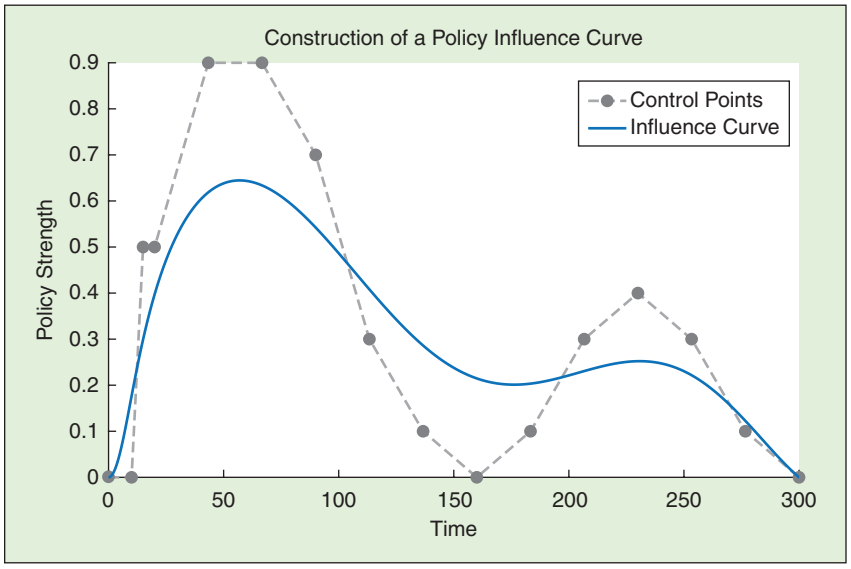


FIGURE 2 A possible policy influence curve showing the underlying control points, through which the curve is defined.

IV. Experiments

Four different multi-objective optimization algorithms are used in the presented multi-objective approach to find optimal control strategies: *NSGA – II* [18], *MOEA/D* [19], as well as *GLMO* [20] in two different configurations: once with *NSGA – II* and once with *NSGA – III* [21]. *GLMO* is a large-scale multi-objective optimization algorithm, which is useful for this study because it deals with a large-scale problem—a maximum of 130 decision variables in the largest experiment.

The presented approach is compared with the model and findings in [7], as both studies use multi-objective optimization to obtain optimal control policies (details in Section IV-B). Table II shows the parameter settings for the control policies.

Each experiment employs a different combination of policies: all possible containment strategies consisting of just a single policy, as well as several strategies with multiple policies (see Table I). The five experiments with multiple policies are as follows: social distancing and lock down measures (*sd.+ld.*), NPIs, PIs, the combination of NPIs and PIs, and the combination of NPIs, PIs, and measures that lessen the negative impact of the pandemic on the economy (abbreviated by ECO). PIs refer to pharmaceutical interventions such as treatment or vaccines, which usually come with a large time delay when compared to NPIs [1], [2]. All experiments use identical initial model parameters (see Table III). The compartment values (S, S_q, E,…) are inherently dynamic, as they are continuously recomputed. The other parameters are constant, unless the containment strategy of a given experiment affects them. In that case, the values are adjusted based on the used policies. All predefined parameters are based on rough estimates of realistic values representing the SARS-CoV-2 pandemic and follow

[7]. Realistic parameter fitting for the current pandemic situation is nontrivial and beyond the scope of this study. Having equal starting conditions, the experiment results differ only in their containment strategies.

The optimization algorithms are run for 10,000 evaluations with a population size of 100. Only the population size for *MOEA/D* had to be set to 91. The simulations start at $t_0 = 0$ and run until $t_n = 300$. The best performing algorithm for each experiment setting is determined by the hyper volume (HV) [22] value and the individual distributions of solutions against the objectives. (1, 1, 1) represents the reference vector for the *HV* calculations, since each objective function has a normalized value between 0 and 1. Each algorithm is run 31 times. For each experiment setting, an aggregated set of non-dominated solutions is determined from the results of each run to provide an in-depth evaluation. A correlation analysis is used to identify the importance of individual parameters.

A. Results

Table IV shows an aggregation of the experiment results. f_1^1 and f_2^1 represent simulations with the highest possible parametric influence ($f_3 = 1$), meaning that the influence curve's height is always 1. This demonstrates the potential of isolated parameter adjustments. Note that this doesn't necessarily correspond to the best result for any of the objectives. The following values represent the median values, including the corresponding standard deviation, for the Pareto front with the highest hyper volume *HV* out of all 31 trials. For each trial, $\widetilde{f_1^{\min}}$ and $\widetilde{f_2^{\min}}$ represent the minimal median value of f_1 or f_2 and $\widetilde{HV}$ represents the median *HV*. For f_3, the maximum value $\widetilde{f_3^{\max}}$ is of interest, since the minimum value of $f_3 = 0$ is a representation of solutions in E_{no}. The last column ($\widetilde{f_3^{\max'}}$) displays the values from $\widetilde{f_3^{\max}}$ multiplied by the number of policies contained in the strategy. For instance, the experiment setting E_{NPI} represents strategies with four policies, hence $\widetilde{f_3^{\max'}} = 4 \cdot \widetilde{f_3^{\max}}$. This presents an alternate view in which the assumed cost of the employed strategy scales with the number of parameter adjustments used. Regarding the experiments on this new model, there are three trends to observe: first, even adjusting single parameters can have major effects. Optimizing c_r, p_{qr}, or c_{dp} produces high *HV* values, which can be an indicator for the importance of *NPIs* in general. Second, using non-NPI related policies without simultaneously employing NPI policies has a very limited effect. It is observable that either the first infection wave hits the population before effective PIs are available or that the vaccine

TABLE I Combinations of parameter adjustments.

SYMBOL	PARAMETER ADJUSTMENTS	DECISION VARIABLES
sd.+ld.	c_r & p_{qr}	26
NPI	c_r, p_{qr}, c_{dp} & p_{qer}	52
PI	d_r, i_{qrr} & v_r	39
NPI + PI	d_r, i_{qrr}, v_r, c_r, p_{qr}, c_{dp} & p_{qer}	91
NPI + PI + ECON	d_r, i_{qrr}, v_r, c_r, p_{qr}, c_{dp}, p_{qer}, $(GDP)p_i$, $(GDP)p_{qi}$ & $(GDP)p_{eqi}$	130

TABLE II Boundaries for the adjustment of parameters (LB [lower boundary], UB [upper boundary], TT [trigger time]).

POLICY/PARAMETER	SYMBOL	LB	UB	TT
Contact Rate	c_r	−7*	0	0
Preventive Quarantine Rate	p_{qr}	0	0.5	0
Contact Detection Probability	c_{dp}	0	0.6	0
Diagnosis Rate	d_r	0	2/14	0.05
Infected Quarantined Recovery Rate	i_{qrr}	0	1/42	0.05
Vaccination Rate	v_r	0	1/100	0.25
Preventive Quarantine End Rate	p_{qer}	0	1/14	0
(GDP) Pandemic Influence	$(GDP)p_i$	−0.03	0	0.1
(GDP) Preventive Quarantine Impact	$(GDP)p_{qi}$	−0.2	0	0.1
(GDP) Exposed Quarantined Impact	$(GDP)p_{eqi}$	−0.2	0	0.1

* Negative, since c_r enforces contact reduction between individuals in absolute values.

TABLE III Model parameters.

PARAMETER	VALUE
Extended SEIR Model Parameters	
Initial S	0.98
Initial E	0.01
Initial I	0.01
Initial S_q, E_q, I_q, R, R_q, D	0
Contact Rate c_r*	10
Transmission Probability t_p	0.05225
Incubation Rate i_r	1/7
Preventive Quarantine Rate p_{qr}*	0
Preventive Quarantine End Rate p_{qer}*	1/14
Contact Detection Probability c_{dp}*	0
Diagnosis Rate d_r *	1/14
Infection Recovery Rate i_{rr}	1/14
Infection Quarantined Recovery Rate i_{qrr}*	1/14
Immunity Loss Rate i_{lr}	1/90
Vaccination Rate v_r*	0
Death Probability d_p	0.01
Economy Model Parameters	
Initial GDP	0
Baseline Growth b_g	0.019346
Pandemic Influence p_i*	0.12
Preventive Quarantined Impact p_{qi}*	0.4
Exposed Quarantined Impact e_{qi}*	0.4
Infected Impact i_i	0.6
Infected Quarantined Impact i_{qi}	0.8

* parameters that are considered for optimization in the experiments

comes later into play than all other measures. Third, containment strategies consisting of multiple policies generate the solutions with the highest *HV* values. Even policies which had little or no effect on their own can contribute in these scenarios. While the estimated, cost according to f_3, is not necessarily higher for these strategies, $\widetilde{f_3^{\max'}}$ shows that this quickly changes when the objective value is modified to account for the number of parameter adjustments. Therefore, it can be said that the advantages of these strategies would likely come with significantly greater cost.

The footnotes of Table IV highlight the best algorithm per experiment regarding the $\widetilde{HV}$ values. With few exceptions, $GLMO_{NSGA-II}$ performed the best. Further evaluations concentrate on the containment strategies with multiple policies because they produce the highest *HV* values. In-depth analysis of these strategies allows assessing the dynamics between the optimized parameters and their influence on specific objectives. Figure 3 shows the *HV* convergence of all four algorithms, which are used in the simulation experiments. While *NSGA – II*, $GLMO_{NSGA-II}$, and $GLMO_{NSGA-III}$ perform equally well, *MOEA/D* fails to converge to similarly high *HV* values. However, it performs reasonably when optimizing parameter adjustments that influence only two of three objectives, e.g., $E_{i_{qrr}}$. Using Figure 4, it can be observed that *MOEA/D* gets stuck locally, since the solutions are dense around specific objective values. The same figure shows, that there is barely a difference between

TABLE IV The containment strategies and corresponding performance indicators. Table footnotes show the best algorithm according to the HV values.

	EXPERIMENT	f_1^1	f_2^1	$\widetilde{HV}$	$\widetilde{f_1^{\min}}$	$\widetilde{f_2^{\min}}$	$\widetilde{f_3^{\max}}$	$\widetilde{f_3^{\max'}}$
	E_{no}	0.4975	−3.0301	-	-	-	-	
NPIs	E_{c_r} [b]	0.1960	**−4.9158**	0.5135 ± 0.0312	0.1088 ± 0.0051	−4.9669 ± 0.0368	0.8835 ± 0.0156	
	$E_{p_{qr}}$ [d]	0.0765	6.3634	0.6238 ± 0.0011	0.0599 ± 0.0000	−3.0301 ± 0.0000	**0.3095** ± 0.0312	
	$E_{c_{dp}}$ [d]	0.1440	0.9980	0.5653 ± 0.0007	0.1115 ± 0.0001	−3.0301 ± 0.0000	0.8171 ± 0.0715	
	$E_{p_{qer}}$ [–]	0.4975	−3.0301	0.0000 ± 0.0000	0.4975 ± 0.0000	−3.0301 ± 0.0000	0.0000 ± 0.0000	
PIs	E_{d_r} [c]	0.4371	−3.8430	0.1462 ± 0.0035	0.4025 ± 0.0001	−3.8099 ± 0.0142	0.8860 ± 0.0193	
	$E_{i_{qrr}}$ [b]	0.4882	−3.3365	0.0225 ± 0.0000	0.4836 ± 0.0000	−3.2825 ± 0.0151	0.6205 ± 0.0622	
	E_{v_r} [b]	0.4975	−3.5046	0.0000 ± 0.0000	0.4975 ± 0.0000	−3.4528 ± 0.0178	0.6887 ± 0.0396	
ECON	$E_{(GDP)p_i}$ [b]	0.4975	−3.5543	0.0000 ± 0.0000	0.4975 ± 0.0000	−3.5041 ± 0.0322	0.7454 ± 0.0753	
	$E_{(GDP)p_{qi}}$ [–]	0.4975	−3.0301	0.0000 ± 0.0000	0.4975 ± 0.0000	−3.0301 ± 0.0000	0.0000 ± 0.0000	
	$E_{(GDP)p_{eqi}}$ [–]	0.4975	−3.0301	0.0000 ± 0.0000	0.4975 ± 0.0000	−3.0301 ± 0.0000	0.0000 ± 0.0000	
COMBINATIONS	$E_{\text{sd.+ld.}}$ [d]	0.0649	6.3466	0.7863 ± 0.0056	0.0509 ± 0.0001	−4.9526 ± 0.0579	0.4629 ± 0.0197	0.9258
	$E'_{\text{sd.+ld.}}$ [7]*	-	-	0.4201 ± 0.0000	0.1765 ± 0.0000	−3.4299 ± 0.0000	0.2545 ± 0.0000	0.5090
	E_{NPI} [d]	**0.0627**	5.0642	0.8519 ± 0.0059	**0.0496** ± 0.0004	−4.9498 ± 0.0359	0.5642 ± 0.0433	2.2568
	$E_{\text{NPI+ECON}}$ [d]	**0.0627**	−0.8341	0.8828 ± 0.0085	**0.0497** ± 0.0003	−4.9930 ± 0.0539	0.5223 ± 0.0637	3.6561
	$E_{\text{NPI+PI}}$ [d]	**0.0627**	5.0555	0.9141 ± 0.0039	**0.0496** ± 0.0001	**−5.4874** ± 0.0254	0.4857 ± 0.0492	3.3999
	$E_{\text{NPI+PI+ECON}}$ [d]	**0.0627**	−0.8428	**0.9261** ± 0.0058	**0.0497** ± 0.0002	**−5.4958** ± 0.0314	0.4859 ± 0.0548	4.8590

[a] *NSGA – II*.
[b] *MOEA/D*.
[c] *GLMO* with *NSGA – III*.
[d] *GLMO* with *NSGA – II*.
[–] No optimization.
* Reference experiment based on [7].

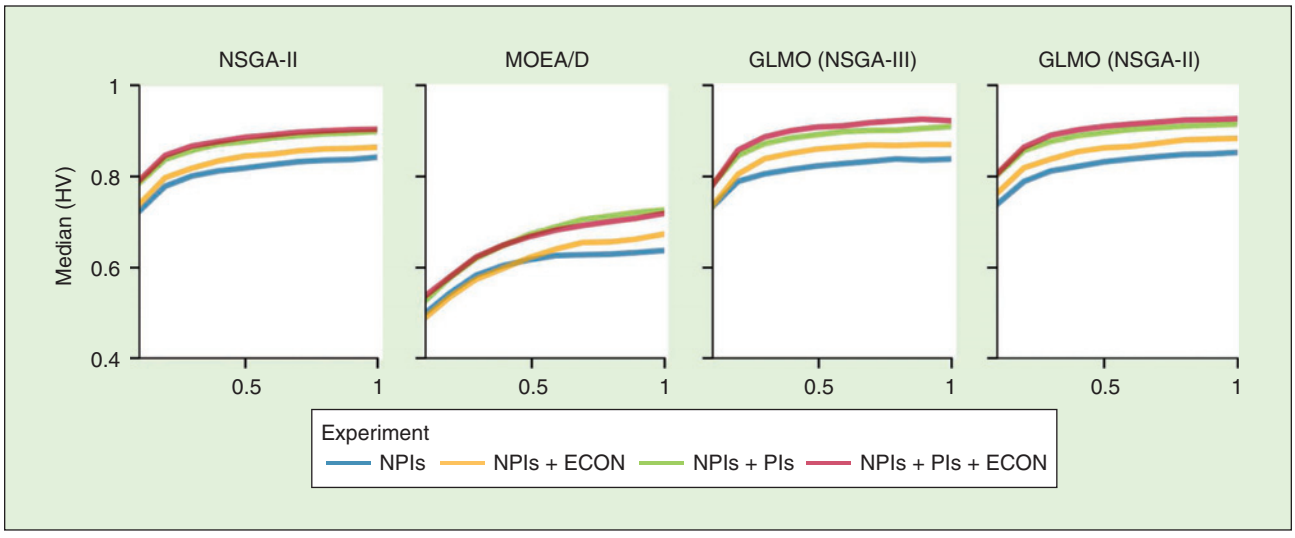


FIGURE 3 Hyper volume convergence.

$GLMO_{NSGA-II}$ and $GLMO_{NSGA-III}$. Finally, it is visible that $NSGA-II$ performs similarly to the corresponding $GLMO$ implementation which uses $NSGA-II$ internally.

B. Comparison

The experiment results are compared with the work in [7], which optimizes only two policies. The result is indicated as $E'_{sd.+ld.}$ in Table IV. For this experiment, identical initial parameters for both the policies and optimization were used. The definition of policies as outlined in [7] remains unchanged, and the optimizer operates on the trigger times. However, the optimizer is adjusted to respect the same three objectives as presented in this paper. Figure 5 visualizes the comparison between $E_{NPI+PI+ECON}$ and $E_{sd.+ld}$. The proposed approach offers a multitude of possible containment strategies, while [7] only provides solutions on a linear scale without variations in the measures' strength. Evidently, the rigid approach of [7] limits the possibilities for optimization, as the obtained solutions are limited in performance regarding the objectives and the coverage of the objective space itself.

C. Influence of Parameters

Due to their inherent nature, certain model parameters are more influential than others, e.g., c_r or t_p. The potential of strategies is further guided by the limits for each parameter during the optimization (see Table II). For this reason, the discussion concentrates mainly on the general dynamics illustrated by the pandemic model. The parameters with high influence are detected using a correlation coefficient and regression analysis (see Figure 6) and by looking at the

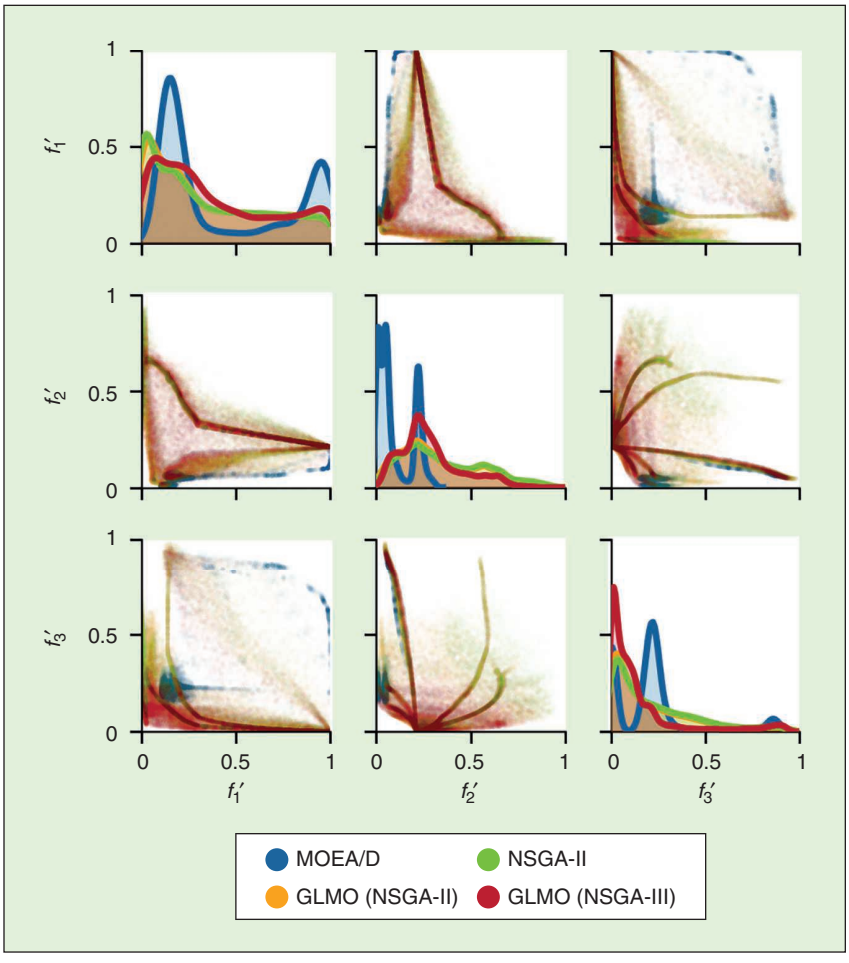


FIGURE 4 The scatter plots visualize the aggregated Pareto fronts over 31 runs.

single parametric experiment runs (see Table IV). Figure 5 shows the Pareto front for $E_{NPI+PI+ECON}$, which is the experiment setting with the highest overall *HV*. It can be observed, that manipulating p_{qr} (e.g., a lockdown) positively correlates with f_2, damaging the economy. On the other hand, adjustments to c_r is beneficial to both main objectives. Exploring correlations with f_3 can be used as a proxy for the strength of a parameter's influence. For instance, p_{qr} has high costs on economy and simultaneously low correlation with the f_3 values, which suggests that the optimized solutions avoid heavy use of this parameter. In general, solutions which favor f_2 tend to be good trade-off solutions. The best solutions regarding f_1 sacrifice f_2 to a considerable degree, but accepting a slightly worse f_1 makes it possible to optimize f_2 to a high degree alongside it. This stands out when comparing the positions of the extreme solutions for f_1 and f_2, where those for f_2 tend to be close to the prominent knee point of the Pareto fronts, while those for f_1 tend to be on the far end of the spectrum. The cost of the strategies (f_3) seems symmetrically distributed, which means that the mentioned dynamic also translates to cheaper solutions. It is evident that getting close to optimal f_1 or f_2 values is possible with little cost, as indicated by the light colored solutions that extend to the borders of the estimated Pareto fronts (see Figure 5). However, the less costly a containment strategy is, the stronger is its inherent trade-off. If both objectives are to be optimized simultaneously, increased costs are inevitable. This result suggests that solving the HED dilemma successfully for health and economy goals requires high intervention efforts.

D. Containment Strategies

The extreme points in E_{NPI} can be used to illustrate possible decision-making strategies (see Figure 6). More complex experiments can achieve better results, but a simpler setting aids the identification of possible strategies. It is problematic to nominate a single best strategy, since the presented use case does not refer to a real-world scenario and the decision-makers' preferences are unknown. The solution causing the lowest infection peaks ($f_1 = 0.0497$) modifies mainly three of four available parameters. Adjustments to p_{qer} (e.g., the length of preventive quarantine) are largely neglected in this strategy. This strategy has very pronounced oscillations, with parameters non-overlapping during their dominant phases. The average cost of the strategy (f_3) is below 0.45. This is comparable to the second strategy (see Figure 6(b)), which is optimized for economic well-being. In this case, the optimizer only utilized two parameters: c_r and p_{qer}. Surprisingly, the latter only builds up to full strength after the first infection wave has already faded. The third strategy in this figure represents the one which utilizes the available policies to the greatest extent ($f_3 = 0.68$). From one perspective, it seems to be an inefficient version of the health-optimal strategy, which achieves the same minimal infection numbers at lower cost. Upon closer inspection, it is evident that the increased cost has been utilized to improve upon the economy objective without sacrificing the obtained health objective. Without tracking costs, this strategy would appear to be strictly superior to the first, when it is, in fact, a trade-off.

E. Statistical Evaluation

All trials are grouped by each experiment setting E (see Figure 4) and by the algorithms that were used on them in the optimization process. This allows us to assess the performances of the algorithms in comparison with each other, while being able to identify effective experiment settings. Both group configurations are evaluated using a Kruskal–Wallis test, followed by a Mann–Whitney-U test. Containment strategies whose optimized policies do not influence the simulation fail the Kruskal–Wallis test and are not considered for

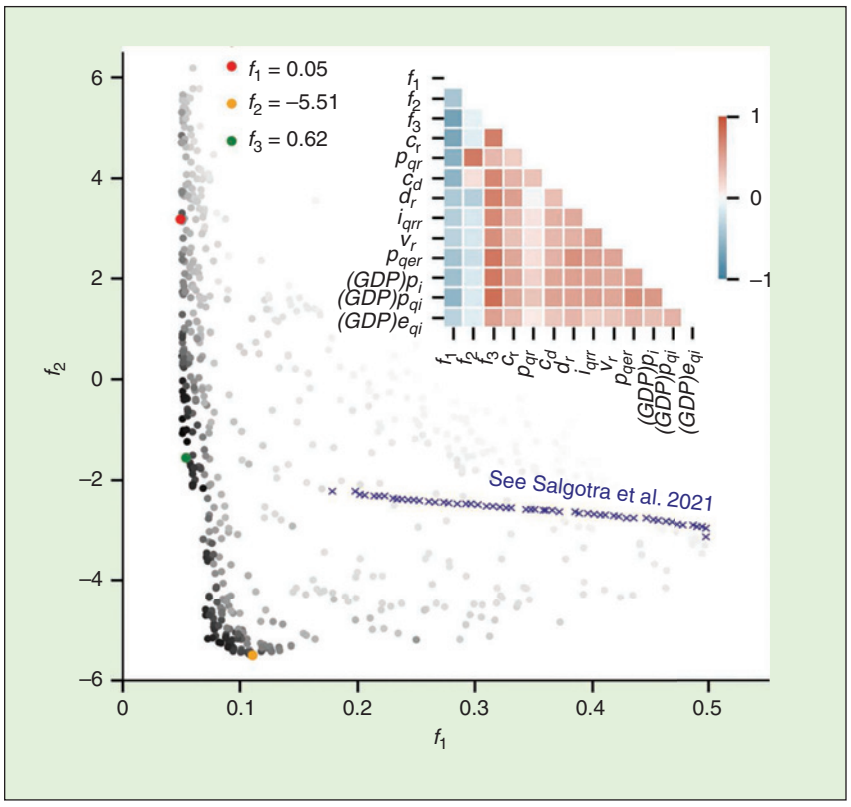


FIGURE 5 Comparison results. Combined Pareto front for $E_{NPI+PI+ECON}$, including the correlation coefficients' matrix. Dark (bright) gray points indicate high (low) f_3 values. The red, orange, and green points show the minimal f_1, f_2 or maximal f_3 values. × shows the results obtained by [7].

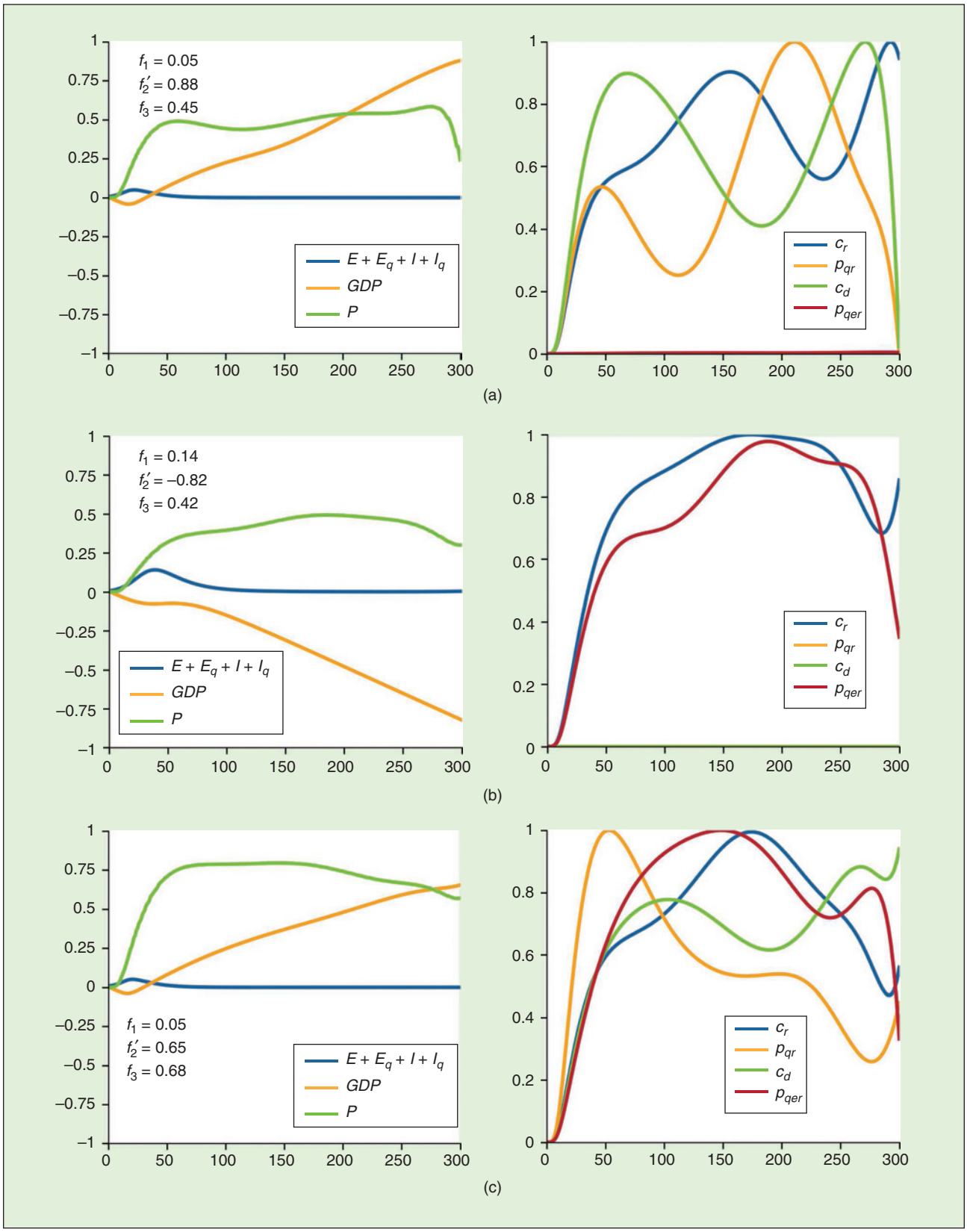


FIGURE 6 Three extreme strategies for E_{NPI}. (a) and (b) represent the strategy with the lowest value of f_1 and f_2' (scaled value for better visualization) over all solutions. (c) represents the strategy with the highest value for f_3, which corresponds to the strategy in which the most intense intervention efforts were made. (a) $f_1^{\min}$ (b) $f_2^{\min}$ (c) $f_3^{\max}$

It is evident that getting close to optimal f_1 or f_2 values is possible with little cost, … However, the less costly a containment strategy is, the stronger is its inherent trade-off.

further evaluations—this study uses a significance level of $p < 0.05$.

The tests show that all experiment settings differ significantly from each other according to their *HV* after the elimination of inconsequential strategies. Although most experiment settings result in differing f_1 values, several of the tested combinations are very similar in this respect. Looking at f_2, it is of interest that $E_{sd.+ld.}$ is highly similar to $E_{NPI}(p = 0.4996)$. Likewise, E_{NPI+PI} is highly similar to ($p = 0.1507$). The comparison regarding f_3 is not necessary because it is only a secondary objective in this model, guiding the algorithms to converge to efficient containment strategies. The next examination tests if the trials with the best *HV* value differ significantly from the other algorithms' trials for each experiment setting. Using *HV* to compare the groups, in E_{ϵ_r} *MOEA/D* is indifferent from $GLMO_{NSGA-III}$ ($p = 0.1432$) and from $GLMO_{NSGA-II}$ ($p = 0.2847$). Testing the f_1 values, all algorithms differ significantly from the algorithm with the best $\widetilde{HV}$ value. Testing f_2 and f_3, the same insights apply in most cases, but some tests fail when comparing $NSGA-II$ and $GLMO_{NSGA-II}$. As such, E_{NPI} ($p = 0.1871$), $E_{p_{qr}}(p = 0.1594)$, and $E_{sd.+ld.}(p = 0.1078)$ do not differ from each other for these two algorithms according to their f_2 values. In case of $E_{NPI+PI}(p = 0.0925)$ and $E_{NPI+ECON}$ ($p = 0.0660$) the values of f_3 show no significant difference.

F. Discussion

This work's approach to finding optimal control strategies during pandemics highlights dynamics inherent to the HED and is suited to support decision-making. However, the presented simulation results do not directly translate to the current pandemic situation. The model parameters must be continuously calibrated to ensure a good fit for a specific pandemic situation for a real-world application. Additionally, parameter limits for the containment measures need to be estimated realistically. The suggested containment strategies need to be interpreted regarding a possible real-world implementation, which raises questions such as what it means to employ a policy at a specific strength.

In this work, the chosen model parameters and boundaries for the optimization experiments have the purpose to demonstrate this novel approach, to identify general correlations within the model, and to see where parallels and general insights can be found towards real-world scenarios. The length of the simulated pandemic is limited to a specific timeframe. It should be kept in mind that this can bias the results to favor solutions optimized for the chosen timeframe. In Figure 6(a) for example, it seems that the end of two of the three employed policies coincides perfectly with the end of the simulated timeframe. This reduces the cost objective of the strategy without the consequences of abruptly terminating such measures. As the pathogen requires time to build up to another wave, this wave never manifests due to the cut-off point at the end of the simulation. The results show that NPIs tend to dominate over PIs. During the simulation, it is possible to continuously suppress the virus via NPIs, without ever achieving herd immunity and without having to deal with the long-term consequences of that. Measures like the vaccine with long-lasting effects could be more cost-effective over longer timeframes. That said, there is another reason why NPIs might be favored by the optimization: most NPIs influence a significantly greater portion of the population, namely the healthy population, than PIs, which mostly target the much smaller fraction of already infected individuals. Hence, NPIs have a leverage advantage due to the higher number of individuals with which they can effectively work. Future studies on integrated strategies with PIs and NPIs can improve upon the presented approach by considering longer timeframes or indicators for herd immunity as a termination criterion.

V. Conclusion

This paper explores novel pandemic containment strategies regarding three conflicting objectives. Implementing multi-objective optimization allows the identification of proper containment strategies, producing minimal infection peaks and minimal economic damage, while having low overall cost in terms of regulatory influence. The experiments are based on an extended version of the *SEIR* compartmental model, with an integrated economic compartment and a model for dynamic containment strategies. The results indicate that there are significant trade-offs between the three objectives to be optimized simultaneously. At least one objective has to be sacrificed, since maintaining both a stable economy and health-care system relies on significant intervention efforts. In addition, the results suggest that employing economic or pharmaceutical measures without NPIs has limited effects. NPIs have proven to be the most effective and important measures in containing the simulated pandemic. Future works should integrate the parameter estimation ex ante, to validate the model using real-world data. The presented approach can also be extended by working on a dynamic time-horizon for the simulation duration, or by integrating a more comprehensive representation of the economy and its corresponding objective. In conclusion, the multi-objective optimization approach has high potential for detecting optimal and novel containment strategies, although it remains challenging to interpret the results for real-world decision-making.

Acknowledgments

The authors acknowledge the support of The Volkswagen Foundation to carry out this research.

References

[1] S. Flaxman *et al.*, "Estimating the effects of non-pharmaceutical interventions on COVID-19 in Europe," *Nature*, vol. 584, no. 7820, pp. 257–261, 2020.
[2] N. Christakis, *Apollo's Arrow: The Profound and Enduring Impact of Coronavirus on the Way We Live*. Little, Brown, 2020.
[3] M. McKee and D. Stuckler, "If the world fails to protect the economy, COVID-19 will damage health not just now but also in the future," *Nature Med.*, vol. 26, no. 5, pp. 640–642, 2020, doi: 10.1038/s41591-020-0863-y.
[4] R. Miikkulainen, O. Francon, E. Meyerson, X. Qiu, E. Canzani, and B. Hodjat, "From prediction to prescription: evolutionary optimization of non-pharmaceutical interventions in the COVID-19 pandemic," *IEEE Trans. Evol. Comput.*, vol. 25, no. 2, pp. 386–401, Apr. 2021, doi: 10.1109/TEVC.2021.3063217.
[5] V. S. Tseng, J. Jia-Ching Ying, S. T. Wong, D. J. Cook, and J. Liu, "Computational intelligence techniques for combating COVID-19: A survey," *IEEE Comput. Intell. Mag.*, vol. 15, no. 4, pp. 10–22, Nov. 2020, doi: 10.1109/MCI.2020.3019873.
[6] A. Yousefpour, H. Jahanshahi, and S. Bekiros, "Optimal policies for control of the novel coronavirus disease (COVID-19) outbreak," *Chaos, Solitons Fractals*, vol. 136, p. 109,883, Jan. 2020.
[7] R. Salgotra, A. Moshaiov, T. Seidelmann, D. Fischer, and S. Mostaghim, "Optimal control policies to address the pandemic health-economy dilemma," in *Proc. 2021 IEEE Congr. Evol. Comput. (CEC)*, pp. 720–727, doi: 10.1109/CEC45853.2021.9504758.
[8] W. Kermack and A. McKendrick, "A contribution to the mathematical theory of epidemics," in *Proc. Roy. Soc. London Series A, Containing Papers Math. Phys. Character*, vol. 115, no. 772, pp. 700–721, 1927.
[9] A. Mojeeb, I. K. Adu, and C. Yang, "A simple seir mathematical model of malaria transmission," *Asia Res. J. Math.*, pp. 1–27, 2017, doi: 10.9734/ARJOM/2017/37471.
[10] S. Lai *et al.*, "Effect of non-pharmaceutical interventions to contain COVID-19 in China," *Nature*, vol. 585, no. 7825, pp. 410–413, 2020.
[11] S. He, Y. Peng, and K. Sun, "SEIR modeling of the COVID-19 and its dynamics," *Nonlinear Dyn.*, vol. 101, no. 3, pp. 1667–1680, 2020, doi: 10.1007/s11071-020-05743-y.
[12] C. Colas *et al.*, "EpidemiOptim: A toolbox for the optimization of control policies in epidemiological models," 2020, *arXiv: 2010.04452*.
[13] H. Wulkow, T. O. Conrad, N. D. Conrad, S. A. Müller, K. Nagel, and C. Schütte, "Prediction of COVID-19 spreading and optimal coordination of countermeasures: From microscopic to macroscopic models to Pareto fronts," *PLoS One*, vol. 16, pp. 1–29, Apr. 4, 2021, doi: 10.1371/journal.pone.0249676.
[14] N. Perra, "Non-pharmaceutical interventions during the COVID-19 pandemic: A review," *Phys. Rep.*, vol. 913, pp. 1–52, 2021, doi: 10.1016/j.physrep.2021.02.001.
[15] J. Panovska-Griffiths, C. Kerr, W. Waites, and R. Stuart, "Mathematical modeling as a tool for policy decision making: Applications to the COVID-19 pandemic," pp. 291–326, Jan. 2021.
[16] H. Rahmandad and J. Sterman, "Heterogeneity and network structure in the dynamics of diffusion: Comparing agent-based and differential equation models," *Manage. Sci.*, vol. 54, no. 5, pp. 998–1014, 2008, doi: 10.1287/mnsc.1070.0787.
[17] J. M. Carcione, J. E. Santos, C. Bagaini, and J. Ba, "A simulation of a COVID-19 epidemic based on a deterministic seir model," *Frontiers Public Health*, vol. 8, 2020, doi: 10.3389/fpubh.2020.00230.
[18] K. Deb, A. Pratap, S. Agarwal, and T. Meyarivan, "A fast and elitist multiobjective genetic algorithm: NSGA-II," *IEEE Trans. Evol. Comput.*, vol. 6, no. 2, pp. 182–197, Apr. 2002, doi: 10.1109/4235.996017.
[19] Q. Zhang and H. Li, "MOEA/D: A multiobjective evolutionary algorithm based on decomposition," *IEEE Trans. Evol. Comput.*, vol. 11, no. 6, pp. 712–731, Dec. 2007.
[20] H. Zille, "Large-scale multi-objective optimisation: New approaches and a classification of the state-of-the-art," Ph.D. dissertation, 2019.
[21] K. Deb and H. Jain, "An evolutionary many-objective optimization algorithm using reference-point-based nondominated sorting approach, Part I: Solving problems with box constraints," *IEEE Trans. Evol. Comput.*, vol. 18, no. 4, pp. 577–601, Aug. 2014, doi: 10.1109/TEVC.2013.2281535.
[22] S. Huband, P. Hingston, L. While, and L. Barone, "An evolution strategy with probabilistic mutation for multi-objective optimisation," in *Proc. 2003 Congr. Evol. Comput., CEC '03*, vol. 4, pp. 2284–2291.

Conference Calendar

Marley Vellasco
Pontifícia Universidade Católica do Rio de Janeiro, BRAZIL
Leandro Minku
University of Birmingham, UK

* Denotes a CIS-Sponsored Conference
Δ Denotes a CIS Technical Co-Sponsored Conference

*** 2022 IEEE International Conference on Computational Intelligence in Bioinformatics and Computational Biology (IEEE CIBCB 2022)**
August 15–17, 2022
Place: Ottawa, Canada
General Co-Chair: Dan Ashlock
Website: https://cmte.ieee.org/cis-bbtc/cibcb2022/

*** 2022 IEEE Conference on Games (IEEE CoG 2022)**
August 21–24, 2022
Place: Beijing, China
General Chairs: Dongbin Zhao and Simon M. Lucas
Website: https://ieee-cog.org/2022/

Δ The 4th International Conference on Industrial Artificial Intelligence (IAI 2022)
August 24-27, 2022
Place: Shenyang, China
General Chair: Tianyou Chai
Website: http://journal13.magtech.org.cn/iai2022/

Δ 4th International Conference on Data Intelligence and Security (ICDIS 2022)
August 24-26, 2022
Place: Shenzhen, China
General Chairs: Yaochu Jin and Xuan Wang
Website: https://www.icdis.org/

Δ Second International Conference on Emerging Techniques in Computational Intelligence (ICETCI)
August 25–27, 2022
Place: Hyderabad, India
General chairs: Nikhil R. Pal and Akira Hirose
Website: https://www.ietcint.com/

*** 2022 IEEE International Conference on Development and Learning (IEEE ICDL 2022)**
September 12–15, 2022
Place: London, United Kingdom
General Chairs: Lorenzo Jamone and Yukie Nagai
Website: https://icdl2022.qmul.ac.uk

Δ 7th South-East Europe Design Automation, Computer Engineering, Computer Networks and Social Media Conference (SEEDA-CECNSM 2022)
September 23–25, 2022
Place: Preveza, Greece
General Chairs: Alexandros T. Tzallas, Markos G. Tsipouras, and Michael F. Dossis
Website: https://seeda2022.conf.uoi.gr/

Δ 2022 First International Conference on Cyber-Energy Systems and Intelligent Energy (ICCSIE 2022)
October 12–13, 2022
Place: Beijing, China
General Chairs: Huaguang Zhang and Derong Liu
Website: TBA

*** 2022 IEEE International Conference on Data Science and Advanced Analytics (IEEE DSAA 2022)**
October 13–16, 2022
Place: Shenzhen, China
General Chairs: Joshua Huang, Gary Yen, and Karoli Skala
Website: http://dsaa2022.dsaa.co/

Δ 2022 Asian Conference on Artificial Intelligence Technology (ACAIT 2022)
October 28–30, 2022
Place: Changzhou, China
General Co-Chairs: Qionghai Dai, Cesare Alippi, and Jong-Hwan Kim
Website: http://www.acait.cn/

Δ The 17th International Workshop on Semantic and Social Media Adaptation and Personalization (SMAP 2022)
November 3–4, 2022
Place: Virtual
General Co-Chairs: Phivos Mylonas and Katia-Lida Kermanidou
Website: https://hilab.di.ionio.gr/smap2022/

*** IEEE Latin-America Conference on Computational Intelligence (IEEE LA-CCI 2022)**
November 23–25, 2022
Place: Montevideo, Uruguay
General Chairs: Martin Pedemonte and Hector Cancela
Website: https://la-cci.org/

*** 2022 IEEE Symposium Series on Computational Intelligence (IEEE SSCI 2022)**
December 4–7, 2022
Place: Singapore
General Chairs: Ah-Hwee Tan, Dipti Srinivasan and Chunyan Miao
Website: https://www.ieeessci2022.org/

*** 2022 IEEE Smart World Conference (IEEE SWC 2022)**
December 16–18, 2022
Place: Haikou, China
General Chair: Laurence T. Yang
Website: http://www.ieee-smart-world.org/

Digital Object Identifier 10.1109/MCI.2022.3181348
Date of current version: 19 July 2022

IEEE Collabratec™

Bright Minds. Bright Ideas.

Introducing IEEE Collabratec™

The premier networking and collaboration site for technology professionals around the world.

IEEE Collabratec is a new, integrated online community where IEEE members, researchers, authors, and technology professionals with similar fields of interest can **network** and **collaborate**, as well as **create** and manage content.

Featuring a suite of powerful online networking and collaboration tools, IEEE Collabratec allows you to connect according to geographic location, technical interests, or career pursuits.

You can also create and share a professional identity that showcases key accomplishments and participate in groups focused around mutual interests, actively learning from and contributing to knowledgeable communities. All in one place!

Network.
Collaborate.
Create.

Learn about IEEE Collabratec at
ieee-collabratec.ieee.org

IEEE

Advisory Chairs
Bo Xu (Chinese Academy of Sciences, China)
Risto Miikkulainen (University of Texas at Austin, United States)

General Chairs
Dongbin Zhao (Chinese Academy of Sciences, China)
Simon M. Lucas (Queen Mary University of London, United Kingdom)

Program Chairs
Diego Pérez Liébana (Queen Mary University of London, United Kingdom)
Yuanheng Zhu (Chinese Academy of Sciences, China)
Jialin Liu (Southern University of Science and Technology, China)

Local Chair
Qichao Zhang (Chinese Academy of Sciences, China)

Keynote Chairs
Julian Togelius (New York University, United States)
Zongqing Lu (Peking University, China)

Tutorial Chairs
Ruck Thawonmas (Ritsumeikan University, Japan)
Mark Winands (Maastricht University, Netherlands)

Competition Chairs
Xiaochuan Zhang (Chongqing University of Technology, China)
Raluca D. Gaina (Queen Mary University of London, United Kingdom)

Industry Chairs
Quan Yuan (Inspir.ai, China)
Hongliang Li (ByteDance.com, China)

Special Session Chairs
Jianye Hao (Huawei Noah's Ark, China)
Mike Preuss (Leiden University, Netherlands)

Finance Chair
Yaran Chen (Chinese Academy of Sciences, China)

Proceedings Chair
Ding Wang (Beijing University of Technology, China)

Demonstrations Chairs
Xiali Li (Minzu University of China, China)
Haoran Li (Chinese Academy of Sciences, China)

Publicity Chairs
Sanaz Mostaghim (Otto von Guericke University Magdeburg, Germany)
Xiaohan Zhang (Chinese Academy of Sciences, China)
Mark J. Nelson (American University, United States)

Media Chairs
Ning Lu (Chinese Academy of Sciences, China)
Cristiana Pacheco (Queen Mary University of London, United Kingdom)

Diversity & Inclusion Chair
Luntong Li (Chinese Academy of Sciences, China)

Webmaster
Shasha Liu (Chinese Academy of Sciences, China)

Introduction

IEEE Conference on Games (CoG) 2022 will be held from August 21st to 24th, 2022 at Beijing, China. This is the first time for the IEEE CIS flagship conference CoG to come to China, welcomed by the boosting game industry and academia in China and all the world. Games not only establish one of the most profitable industries worldwide, but also offer a general and challenging environment for the advance of Artificial Intelligence (AI) and Computational Intelligence (CI). The annual IEEE Conference on Games (IEEE CoG) brings together leading researchers and practitioners from academia and industry in the field of games to discuss recent advances and explore future directions. It covers all topics in the field of games, from game design to game intelligence and game theory, including scientific, technical, engineering and societal aspects.

Topic (*All aspects of games, including, but not limited to, the following areas*)

- Artificial/Computational Intelligence in Games
  - Deep learning / reinforcement learning / evolutionary computation / fuzzy systems / multi-agent systems / tree and graph search methods / knowledge-based methods / artificial general intelligence in games
  - Real-world problem solving and decision-making
  - Game theory
- Game Design
  - Procedural content generation / Game balance
  - Automatic game design and optimization
  - Mixed-initiative game design
  - Human-AI cooperative creativity
- Game Technologies
  - Multimedia technologies in games
  - Game interfaces and user interaction
  - Virtual and augmented reality
  - Game adaptation and content generation
  - Player modeling
  - Affective modeling and emotion recognition
  - Character development
  - Virtual cinematography
- Game Benchmarks and Competitions
  - Integrated game environment
  - State-of-the-art game AI / General game AI
- Game to Real World
  - Game education / Simulation training

Call for Tutorials

We invite submissions for tutorials to be held at IEEE CoG 2022. This is an opportunity for you to share your expertise and influence future research directions in the CoG community. Tutorials can be on any topic in the scope of the conference.

Call for Competitions

We invite proposals for competitions to be held at the conference. These may be completely new ones or competitions held already in the last years, possibly at other venues. Competitions can be based on well-known games, but competitions based on custom-made and lesser-known games are also welcome.

Call for Special Session Proposals

A special session addresses one or more topic areas within games research and is intended to bring together researchers working on those topics to provide an excellent session at the IEEE CoG 2022. Please read the call for papers for CoG 2022 and its list of topics before submitting your special session proposal.

Call for Industry Talk Proposals

The industry day is an open day with topics provided by and focused on the games industry and how it links to research in games. We welcome talk proposals from the games industry that wish to present their work on the areas covered in this conference. We are interested in presentations about research prototypes, commercial products, indie/mobile/AAA games, AR/VR applications, etc., as well as participation on poster sessions and discussion panels.

Papers and Presentations (*Peer-review is double-anonymous*)

Full technical papers: Full papers have an 8 page limit (including references and appendices), and should constitute a technical or empirical contribution to scientific, technical, and engineering aspects of games.
Short papers (2-4 pages) describe work in progress, smaller projects that are not yet ready to be published as a full paper, or new progress on projects that have been reported elsewhere.
Competition papers (8 pages) describe research related to one of the conference competitions, including the design of new competitions and in particular submissions to existing competitions.
Vision papers (8 pages) describe a vision for the future of the Games field or some part of it, are based on extensive research and include a comprehensive bibliography. Standards for competition papers are as high as for other CoG papers, and standards for vision papers are higher.
Demo papers (2 pages) describe work in progress and will be presented during a demo session.
As with IEEE CoG 2022, we will invite the principal authors to submit an extended version of their papers to the IEEE Transactions on Games (ToG), etc.

Important Dates (*All times are anywhere on Earth, and are firm without extension*)

Competitions, Tutorials, and Special Session Proposals…….. 15th January, 2022
Regular Papers (Full Technical Papers)……... 1st March, 2022
Auxiliary Papers (Short, Competition, Vision, and Demo) ……..14th May, 2022
Travel Grants…..…… 14th May, 2022
Games Industry Talks……... 11th June, 2022
Early bird (and author) registration……... 30th June, 2022

IEEE
IEEE Computational Intelligence Society

Digital Object Identifier 10.1109/MCI.2022.3182849

IEEE SSCI is an established flagship annual international series of symposia on computational intelligence sponsored by the IEEE Computational Intelligence Society to promote and stimulate discussion on the latest theory, algorithms, applications and emerging topics on computational intelligence. By co-locating multiple symposia under one roof, each dedicated to a specific topic in the CI domain, IEEE SSCI aims to encourage cross-fertilization of ideas and provide a unique platform for top researchers, professionals, and students from all around the world to discuss and present their findings. IEEE SSCI 2022 will feature keynote addresses, tutorials, panel discussions and special sessions, all of which are open to all participants. The conference proceedings of the IEEE SSCI will be included in the IEEE Xplore and indexed by all major databases.

List of Confirmed Symposia and Special Sessions

- Adaptive Dynamic Programming and Reinforcement Learning (IEEE ADPRL)
- Artificial Life (IEEE ALIFE)
- Automated Algorithm Design, Configuration and Selection (IEEE AADCS)
- CI in Agriculture (IEEE CIAg)
- CI for Brain Computer Interfaces (IEEE CIBCI)
- CI in Biometrics and Identity Management (IEEE CIBIM)
- Computational Intelligence in Big Data (IEEE CIBD)
- CI in Control and Automation (IEEE CICA)
- CI in Healthcare and E-health (IEEE CICARE)
- CI in Cyber Security (IEEE CICS)
- CI in Data Mining (IEEE CIDM)
- CI in Dynamic and Uncertain Environments (IEEE CIDUE)
- CI and Ensemble Learning (IEEE CIEL)
- CI for Engineering Solutions (IEEE CIES)
- CI for Human-like Intelligence (IEEE CIHLI)
- CI in IoT and Smart Cities (IEEE CIIoT)
- CI for Multimedia Signal and Vision Processing (IEEE CIMSIVP)
- CI in Remote Sensing (IEEE CIRS)
- CI for Security and Defense Applications (IEEE CISDA)
- CI for Astroinformatics (IEEE CIAstro)
- CI in Vehicles and Transportation Systems (IEEE CIVTS)
- Deep Learning (IEEE DL)
- Evolving and Autonomous Learning Systems (IEEE EALS)
- Explainable Data Analytics in Computational Intelligence (IEEE EDACI)
- Evolutionary Neural Architecture Search and Applications (IEEE ENASA)
- Evolutionary Scheduling and Combinatorial Optimisation (IEEE ESCO)
- Ethical, Social and Legal Implications of Artificial Intelligence (IEEE ETHAI)
- CI in Feature Analysis, Selection and Learning in Image and Pattern Recognition (IEEE FASLIP)
- Foundations of CI (IEEE FOCI)
- Immune Computation (IEEE IComputation)
- Intelligent Agents (IEEE IA)
- Model-Based Evolutionary Algorithms (IEEE MBEA)
- Multi-agent System Coordination and Optimization (IEEE MASCO)
- Multicriteria Decision-Making (IEEE MCDM)
- Nature-Inspired Computation in Engineering (IEEE NICE)
- Neuromorphic Cognitive Computing (IEEE SNCC)
- Robotic Intelligence in Informationally Structured Space (IEEE RiiSS)
- Cooperative Metaheuristics (IEEE SCM)
- Differential Evolution (IEEE SDE)
- Swarm Intelligence Symposium (IEEE SIS)

Special Sessions

- Advancing Capabilities of Simulation Models with Computational Intelligence (ASM)
- Artificial Intelligence-based Uncertainty Quantification (AUQ)
- CI in in Soil and Water Management (SWM)
- Evolving Deep and Transfer Learning Models for Computer Vision and Medical Imaging (ECV)
- Games (GAMES)
- Human and Machine Intelligence in Collaborative Decision Making (HMI)
- Conjunction of Quantum Computing and Evolutionary Computation (EVO-QUANTUM)

IEEE SSCI 2022 welcomes your paper submission to the above symposia and special sessions. Proposals for new symposia and special sessions are also welcome. Inquiries and submission of symposium and special session proposals should be addressed to the SSCI 2022 Program Chairs.

Call for Tutorials

IEEE SSCI 2022 solicits proposals for tutorials on specific topics of interests, which will form an integral part of the conference program. Inquiries and submission of tutorial proposals should be addressed to the Keynote/Tutorial Chairs.

Important Dates

Special Session/Tutorial Proposals	Friday, 1st April 2022
Paper Submission Deadline	Friday, 1st July 2022
Notification to Authors	Thursday, 1st September 2022
Full Manuscript Submission	Monday. 19th September 2022
Early Bird Registration	Monday, 26th September 2022
IEEE SSCI 2022 Conference	4th December to 7th December 2022

Organizing Committee

Advisory Chairs

Kay-Chen TAN
City University of Hong Kong

Yew-Soon ONG
Nanyang Technological University

General Chairs

Ah-Hwee TAN
Singapore Management University

Dipti SRINIVASAN
National University of Singapore

Chunyan MIAO
Nanyang Technological University

Program Chairs

Hisao ISHIBUCHI
Southern University of Science and Technology

Chee-Keong KWOH
Nanyang Technological University

Finance Chair

Jian-Chao YAO
DSO National Laboratories

Keynote/Tutorial Chairs

Yaochu JIN
Bielefeld University

Mahardhika PRATAMA
Nanyang Technological University

Exhibit/Competition Chair

Chi-Keong GOH
AI2Labs

Publication Chairs

Anupam TRIVEDI
National University of Singapore

Keeley CROCKETT
Manchester Metropolitan University

Publicity Chairs

Teck-Hou TENG
ST Engineering

Catherine HUANG
McAfee AI Research

Pauline C. HADDOW
Norwegian University of Science and Technology

Jialin LIU
Southern University of Science and Technology

Local Organizing Chairs

Di WANG
Nanyang Technological University

Zhaoxia WANG
Singapore Management University

Hao ZHANG
Nanyang Technological University

Sponsors

IEEE Computational Intelligence Society
IEEE SINGAPORE SECTION
SINGAPORE EXHIBITION & CONVENTION BUREAU
SINGAPORE Passion Made Possible

Organizers

SMU SINGAPORE MANAGEMENT UNIVERSITY
NUS National University of Singapore
NANYANG TECHNOLOGICAL UNIVERSITY SINGAPORE

Digital Object Identifier 10.1109/MCI.2022.3182850