Towards Self-Sufficient High-Rises
Performance Optimisation using Artificial Intelligence

Ekici, B.

DOI
10.7480/abe.2022.10

Publication date
2022

Document Version
Final published version

Citation (APA)
Ekici, B. (2022). *Towards Self-Sufficient High-Rises: Performance Optimisation using Artificial Intelligence*. [Dissertation (TU Delft), Delft University of Technology]. A+BE | Architecture and the Built Environment. https://doi.org/10.7480/abe.2022.10

Towards Self-Sufficient High-Rises

Performance Optimisation using Artificial Intelligence

Berk Ekici

Towards Self-Sufficient High-Rises

Performance Optimisation using Artificial Intelligence

Berk Ekici

A+BE | Architecture and the Built Environment | TU Delft BK

22#10

Design | Sirene Ontwerpers, Véro Crickx

Cover image | Berk Ekici & Véro Crickx

Keywords | High-rise buildings, Self-sufficiency, Energy, Food, Daylight, Performance-based design, Machine learning, Optimisation

ISBN 978-94-6366-562-9
ISSN 2212-3202

This dissertation is open access at https://doi.org/10.7480/abe.2022.10

TOC

Towards Self-Sufficient High-Rises

Performance Optimisation using Artificial Intelligence

Dissertation

for the purpose of obtaining the degree of doctor
at Delft University of Technology,
by the authority of the Rector Magnificus Prof.dr.ir. T.H.J.J. van der Hagen,
chair of the Board for Doctorates
to be defended publicly on
Monday, 27 June 2022 at 15.00 o'clock

by

Berk EKİCİ
Master of Science in Architecture, Yaşar University, Turkey
born in İzmir, Turkey

TOC

This dissertation has been approved by the promotors.

Composition of the doctoral committee:

Rector Magnificus,	Chairperson
Prof.dr.ir. I.S. Sariyildiz	Delft University of Technology, promotor
Prof.dr. M.F. Tasgetiren	Auburn University, promotor
Dr. M. Turrin	Delft University of Technology, copromotor

Independent members:

Prof.dr. P.W.C. Chan	Delft University of Technology
Prof.dr. M. Overend	Delft University of Technology
Dr. C.L. Martin	Delft University of Technology
Dr. F. De Luca	Tallin University of Technology
Prof.dr.ir. P.J.M. van Oosterom	Delft University of Technology, reserve member

I have not failed.
I have just found 10,000 ways that will not work.

Thomas A. Edison

TOC

Acknowledgements

The time has come to write the acknowledgements, maybe the last but the most difficult part of my dissertation. I have gone through the entire process of my Ph.D. journey, which has reminded me of being endorsed by many professors, colleagues, friends, and family. In the end, having the full support of these very people in my life encouraged me all the time to finish what I had started. Maybe my Ph.D. journey ends in Delft, one of the most inspiring university cities on Earth; yet, I know this is just the beginning of my academic career.

Prof.dr.ir. I.S. Sariyildiz, I would like to express my sincere gratitude to my corresponding promotor for accepting me as a Ph.D. candidate in design informatics chair. Prof. Sariyildiz always reminds me of the day when I entered her master class to follow the lectures on parametric design and performance evaluations in 2012. Since then, she has constantly supervised me as a professor and mentor to develop my skills, broaden my perspective, and explore even the most challenging research topics. I am grateful for all her inspirational ideas and encouragement that played an essential role to finish my Ph.D. Most importantly, I am appreciated her endless support in real life that always motivated me to do my best. I will never forget the saying she always reminds us that “knowledge and love increase when shared”.

Prof.dr. M.F. Tasgetiren, I would like to express my sincere gratitude to my promotor for accepting me as a Ph.D. candidate. I remember the day we met to talk about my MSc dissertation. Prof. Tasgetiren immediately denied using available software for artificial intelligence, which encouraged us to code and develop our tools. This vision could have become real only by following his inspirational Ph.D. courses that can teach even the most advanced algorithms to someone holding an architecture diploma. I am grateful for all the support, especially for coding time he never hesitated to share. Besides spending many years together, I appreciate the times outside the faculty when we have met for visiting our imagination to suggest new ideas and concepts. His enthusiasm and curiosity in science and research will always inspire me.

Assoc.prof.dr. M. Turrin, I would like to express my sincere gratitude for being copromotor of my Ph.D. research. If there is a motivational day that can change your life, it is for me the day Dr. Turrin invited me to join one of the lectures of the “Big & Tall” master course at Bouwkunde in 2013. Since then, her motivation and success

TOC

in academic life have influenced me so deeply. I am grateful for all the time she made herself available for discussions and for all those “provocative questions” that assisted me in enlarging my perspective.

I am also profoundly thankful to my committee members, **Prof.dr. P.W.C. Chan**, **Prof. dr. M. Overend**, **Prof.dr.ir. P.J.M van Oosterom**, **Dr. C.L. Martin** and **Dr. F. De Luca** for evaluating my Ph.D. dissertation, giving valuable comments, and spending their valuable time.

Besides the supervisory team and committee members, I sincerely thank **Prof.dr. T. Kazanasmaz,** for all the collaboration and valuable feedback on my research. I would like to thank my friend and Ph.D. fellow **C. Çubukçuoğlu (MSc.)** for her support by spending many hours in the central library and CC-Delft. It was my pleasure to collaborate with you on many papers and share what we had learned! Special thanks go to **Ir. O.F.S.F. Türkcan** for the amazing support during the collaboration, outside of faculty, and helping with the Dutch translation of the summary and propositions. Without incredible support to access many computers in Bouwkunde, I could not realise this research. Therefore, I sincerely thank **Ir. H. Hoogenboom** and **A. Balcı (BSc.)** for their help to conduct my results and insightful talks after work on Thursdays. I would like to present my special thanks to **Dr.ir. I. Chatzikonstantinou**, who patiently responded to my questions over the years, for sharing his profound knowledge on parametric design, performance evaluation, machine learning and optimisation. I would also like to thank **Dr. A. Kırımtat** for all the collaborative works and support. As the first colleague who celebrated the acceptance to my Ph.D. in Sendai/Japan, I thank **Dr. O. Dursun** for guiding me to formulate the research proposal and collaborating on many papers. Additionally, I sincerely thank **Prof.dr.ir. A.A.J.F. van den Dobbelsteen** for valuable suggestions to focus on the Europoint complex as a case study, and the **MOR Team** for providing the materials of the Europoint complex proposal prepared for the Solar Decathlon competition in 2019. Moreover, I would like to thank **Prof.dr. T. Başaran** for all the inspirational lectures on heat transfer in buildings. Finally, I would like to thank **M. Yufka (MSc)** and **Ir. F. Fortich-Mora** for having me on the supervisory team of their master's program. It was my pleasure to share my knowledge while developing the models used in your dissertations.

Apart from that, I owe an appreciation to my colleagues **Dr. S. Aşut**, **Ir. P. de Router**, **Ir. F.J. Gouwetor**, **Dr. Ö. Balaban**, **H.P. Kiksen**, **Melis Baloğlu (M.Sc)** from the chair of design informatics. Without their accompany during the lectures, workshops, coffee breaks, and lunch, my Ph.D. journey could not have been filled with so many good memories and intense discussions.

The entire process would be pretty painful without sharing our experiences with friends and Ph.D. fellows. I feel so lucky to have all these people to share all the struggle and fun together. I have no doubt that there will always be a connection between us even we are all located in different parts of our planet. I would like to thank my Ph.D. fellows **Miktha**, **Ding** and **Frank** from the design informatics chair for sharing the same path with me, exchanging ideas, coffee talks, and having so much of a good time. My special thanks go to **Nick** and **Cynthia** for constantly supporting me inside/outside the faculty, especially for little gatherings near Oostpoort to relax a bit. There would be a huge emptiness if you were not involved in this story. If there is a reason for having great weekends, it can be only because of spending good time with **Utku** and **Manolya**. I appreciate your accompany in every way that considerably helped me while dealing with lockdowns.

I also thank **Arwa**, **Ata**, **Chris**, **Deniz**, **Ehsan**, **Elena**, **Esaar**, **Fatma**, **Hamed**, **Hilmi**, **Kate**, **Mehmet**, **Milad**, **Özlem**, **Pelin**, **Sahar**, **Samantha**, **Sultan**, **Şule** and **Tolga** for all the good memories and their enthusiasm to come together inside/outside the faculty.

Members of TU Delft-Bouwkunde are familiar with joyful lunch gatherings and coffee talks. These gatherings have always been unique because an inspirational, historical, and cutting-edge building hosts us. Without the people, this very building becomes a huge emptiness with no soul, as I witnessed many times during the lockdowns. Therefore, I deeply thank my colleagues **Adriana**, **Ahmed**, **Ameya**, **Amneh**, **Ana**, **Andrea**, **Bruna**, **Dadi**, **Emeline**, **Federica(s)**, **Felippe**, **Filippo**, **Giulia**, **Hedieh**, **Joana**, **Lia**, **Luuk**, **Magda**, **Marco**, **Mohammad**, **Michiel**, **Nan**, **Nima**, **Pedro**, **Prateek**, **Sophia**, **Tamara**, **Tatiana**, **Tess**, **Tiantian**, **Yu-Chou**, **Yvonne**, **Ziead**, and **Zoheir** for all the great time by making this building meaningful with their existence and fellowship. I would like to thank especially **Prof.ir. P.G. Luscuere** for hosting the barbecue gatherings of our section at his lovely place, where I have some of my best memories of this journey. Special thanks go to **Hamza** for preparing amazing coffee every Friday!

Moreover, a great team always remained available to help all the members of the Architectural Engineering and Technology department. I would like to thank **Prof. dr.ir. M.T. Kreutzer**, **Dr. W.O. de Wit**, **Barbara**, **Bo**, **Danielle**, **Francoise, Linda, Tessa,** and **Yvonne** for all the support and assistance over the years**.** Additionally, I would like to thank **Inge** and **Nouzha** from the faculty graduate school for their assistance and patience. Special thanks to the lecturers and employees of the **University Graduate School**, **Human Resources**, and **International Office** for all their great effort to support us all the time. I owe more than an acknowledgment to **Véro Crickx** for her great support while preparing my Ph.D. book with valuable input and effort in arranging figures and designing the flowchart of the dissertation.

The story of my academic career did not start in Delft. Therefore, it is impossible to finalise my acknowledgements without thanking my professors, colleagues, and friends in İzmir. I am profoundly thankful to **Assoc.prof.dr. R. Polat, Assist.prof. dr. İ. Zincir, Miray & Hüseyin Arslan** for always supporting me, especially during the most difficult times of my research. I would also like to thank **Prof.dr. H.M. Gürel** and **Assoc.prof.dr. A. Yılmaz** for giving me a great opportunity as an adjunct instructor at Yaşar University. I sincerely thank the project leaders, **Assoc.prof.dr. B. Kundakçı-Koyunbaba**, **Assist.prof.dr. S. Kutucu** and **Assist.prof.dr. İ. Kahraman** for having me in their funded research projects and sharing their thoughts about my Ph.D. research. Finally, yet importantly, I would like to thank my office mates and fellows **Berfin**, **Ece**, **Eda**, **Feyza**, **Fulya**, **Gökçe**, **Selin**, **Yarkın** and **Yasemin** for such a great time. I know that most of you have almost completed your research. I wish you the best in the next step of your career! Thanks a lot for your positive attitude in every way that felt me too much that I am part of a big family.

Apart from all, someone very special encouraged me whatever happened during the most challenging times. I did not need to be with her in İzmir to know and feel that I was greatly supported. I present my deepest heart-warming thanks to my girlfriend, **Burçin Alkan**. I am grateful that you were always with me with your trust, patience, and love. I hope the end of this Ph.D. will initiate new beginnings in our future. As we always say, everything will be just great!

Last but not least, I would like to express my sincere gratitude to my family; my dear father **Bülent**, my beloved mother **Dilara**, and my dearest sister **Pınar**. Indeed, there is no way to thank you for all the generosity, patience, encouragement, support, and love. I know very well how much you were with me in every part of this journey. No one else on Earth would like to witness my Ph.D. defence more than three of you. I have no words to thank you for everything, but; I hope you are proud of me, and I dedicate this Ph.D. dissertation to you.

Thank you – Dank u Wel - Teşekkürler

TOC

to Bülent, Dilara and Pınar

Contents

List of Tables

List of Figures

List of abbreviations and symbols

Abbrev. / symbol	Description	Unit
Machine learning		
AI	Artificial intelligence	
ANN	Artificial neural network	
CV	Cross-validation	
DNN	Deep neural network	
FNN	Feedforward neural network	
GBM	Gradient boosting model	
GLM	Generalised linear model	
kNN	k-nearest neighbour	
MAE	Mean absolute error	
MARS	Multivariate adaptive regression splines	
ML	Machine learning	
MLR	Multiple linear regression	
MRA	Multiple regression analysis	
MSE	Mean squared error	
RBF	Radial basis function	
RBFNN	Radial basis function neural network	
ReLU	Rectified linear units	
RF	Random forest	
R^2	R-squared	
SM	Surrogate model	
Std	Standard deviation	
SVM	Support vector machine	

>>>

TOC

Abbrev. / symbol	Description	Unit
Optimisation		
avg f(x)	Average of function x	
CMA-ES	Covariance matrix adaptation with evolution strategy	
CPU	Computation time	
DV	Decision variable	
EC	Evolutionary computation	
FES	Function evaluations	
FES/CPU	The number of completed function evaluations	[1 s]
F_{sph}	Sphere function	
F_{ros}	Rosenbrock's function	
F_{ack}	Ackley's function	
F_{grw}	Griewank's function	
F_{ras}	Rastrigin's function	
F_{sch}	Generalised schwefel's problem 2.26	
F_{sal}	Salomon's function	
F_{wht}	Whitely's function	
F_{pn1}	Generalised penalised function 1	
F_{pn2}	Generalised penalised function 2	
F_1	Shifted sphere function	
F_2	Shifted schwefel's problem 1.2	
F_3	Shifted rotated high conditioned elliptic function	
F_4	Shifted schwefel's problem 1.2 with noise in fitness	
F_5	Schwefel's problem 2.6 with global optimum on bounds	
F_6	Shifted rosenbrock's function	
F_7	Shifted rotated griewank's function without bounds	
F_8	Shifted rotated ackley's function with global optimum on bounds	
F_9	Shifted rastrigin's function	
F_{10}	Shifted rotated rastrigin's function	
GA	Genetic algorithm	
GD	Gradient descent	
HypE	Hypervolume-based many-objective optimisation	
HGPSPSO	Hybrid generalised pattern search particle swarm optimisation	
jEDE	Self-adaptive differential evolution with the ensemble of mutation strategies	
MACO	Multi-objective ant colony optimisation	
max f(x)	Maximum of function x	
min f(x)	Minimum of function x	
MOEA/D	Multi-objective evolutionary algorithm based on decomposition	
NSGA-II	Non-dominated sorting genetic algorithm II	
NSPSO	Non-dominated sorting particle swarm optimisation	
P&S	Population and swarm	
PSO	Particle swarm optimisation	

>>>

Abbrev. / symbol	Description	Unit
Optimisation		
RBFopt	Radial basis function optimisation	
RBFMopt	Multi-objective radial basis function optimisation	
SGD	Stochastic gradient descent	
SPEA-2	Strength pareto evolutionary algorithm 2	
std f(x)	Standard deviation of function x	
NFL	No free lunch	
NFT	Near feasibility threshold	
Performance assessment		
ASE	Annual sunlight exposure	[%]
BIPV	Building-integrated photovoltaic	
DA	Daylight autonomy	[%]
DF	Daylight factor	[%]
DGP	Daylight glare probability	
E_c	Cooling consumption	[MWh]
E_{eq}	Equipment consumption	[MWh]
E_F	Farming energy consumption	[MWh]
E_g	Energy generation	[MWh]
E_h	Heating consumption	[MWh]
E_L	Lighting consumption	[MWh]
E_R	Residential energy consumption	[MWh]
E_{tot}	Total energy consumption	[MWh]
F_p	Food production	[ton]
g&s	Germination and seeding	
g-val.	Solar transmittance of the materials used in simulation models	
IL	Illumination level	[lux]
PV	Photovoltaic	
sDA	Spatial daylight autonomy	[%]
Tvis	Visible transmittance of the materials used in simulation models	
UDI	Useful daylight illuminance	[%]
U-val.	Thermal transmittance of the materials used in simulation models	[W/ m^2 K]
Others		
CTBUH	Council on Tall Buildings and Urban Habitat	
GH	Grasshopper 3D algorithmic modelling environment	
IEA	International Energy Agency	
IES	Illuminating Engineering Society	
LEED	Leadership in energy and environmental design	
MUZO	Multi-zone optimisation	
PCA	Performative computational architecture	
USGBC	United States Green Building Council	

Propositions

accompanying the dissertation

Towards Self-Sufficient High-Rises

Performance Optimisation using Artificial Intelligence

By Berk Ekici

1 The next generation high-rises should provide essential resources in addition to dense habitation in metropoles for a sustainable future.

[This proposition pertains to this dissertation]

2 Performance optimisation of self-sufficient high-rises is complex due to large numbers of design parameters and their interactions (interrelations) besides extensive simulations.

[This proposition pertains to this dissertation]

3 Different floor levels of a high-rise in dense urban areas require various design decisions to achieve well-performing alternatives.

[This proposition pertains to this dissertation]

4 The role of AI in designing self-sufficient high-rises is creating and assessing a considerable number of design alternatives during the conceptual design phase.

[This proposition pertains to this dissertation]

5 Intelligent systems cannot be created without intelligent humans who are inspired by nature.

6 Next-generation architects will have to learn the fundamentals of artificial intelligence and the use of computation power by coding to cope with the design complexity.

7 PhD candidates should be frequently involved in the bachelor's and master's courses to share their findings with students.

8 Lifelong learning through social interaction is faster than reading the literature.

9 "Imagination is more important than knowledge" (Albert Einstein). The increase of knowledge can stimulate the imagination.

10 Borders exist only on the map, not on the land.

These propositions are regarded as opposable and defendable, and have been approved as such by the promotors Prof.dr.ir. I.S. Sariyildiz, Prof.dr. M.F. Tasgetiren, and copromotor Dr. M. Turrin.

TOC

Stellingen

bij de dissertatie

Naar zelfvoorzienende hoogbouw

Prestatie-optimalisatie met behulp van kunstmatige intelligentie

Door Berk Ekici

1 De volgende generatie hoogbouw moet voorzien in essentiële hulpbronnen naast dichte bewoning in metropolen voor een duurzame toekomst.

[Deze stelling heeft betrekking op dit proefschrift]

2 Prestatieoptimalisatie van zelfvoorzienende hoogbouw is complex door grote aantallen ontwerpparameters en hun interacties (onderlinge verbanden) naast uitgebreide simulaties.

[Deze stelling heeft betrekking op dit proefschrift]

3 Verschillende vloerniveaus van een hoogbouw in dichtbevolkte stedelijke gebieden vereisen verschillende ontwerpbeslissingen om tot goed presterende alternatieven te komen.

[Deze stelling heeft betrekking op dit proefschrift]

4 De rol van AI bij het ontwerpen van zelfvoorzienende hoogbouw is het creëren en beoordelen van een aanzienlijk aantal ontwerpalternatieven tijdens de conceptuele ontwerpfase.

[Deze stelling heeft betrekking op dit proefschrift]

5 Intelligente systemen kunnen niet worden gecreëerd zonder intelligente mensen die zich laten inspireren door de natuur.

6 De volgende generatie architecten zal de grondbeginselen van kunstmatige intelligentie en het gebruik van rekenkracht door codering moeten leren om de ontwerpcomplexiteit aan te kunnen.

7 Promovendi moeten veelvuldig betrokken worden bij de bachelor- en masteropleidingen om hun bevindingen met studenten te delen.

8 Een leven lang leren door sociale interactie gaat sneller dan het lezen van de literatuur.

9 “Verbeelding is belangrijker dan kennis” (Albert Einstein). De toename van kennis kan de verbeelding stimuleren.

10 Grenzen bestaan alleen op de kaart, niet op het land.

Deze stellingen worden als tegenstelbaar en verdedigbaar beschouwd, en zijn als zodanig goedgekeurd door de promotoren Prof.dr.ir. I.S. Sariyildiz, Prof.dr. M.F. Tasgetiren, en copromotor Dr. M. Turrin.

Summary

Population growth and urbanisation trends bring many consequences related to the increase in global energy consumption and CO_2 emissions and decrease in arable land per person. Alternative design proposals for sustainable living are on the agenda of researchers and professionals to respond to the needs of the 21st century for a sustainable future. Since the early examples in the 19th century, the high-rises have been one of the inevitable buildings of metropolises to provide extra floor space in compact cities. Based on the facts of the 21st century, high-rise buildings should fulfil more than provide extra floor space in the limited urban plot. This research suggests "*self-sufficient high-rise buildings*" that can generate and efficiently consume vital resources in addition to dense habitation for sustainable living. Optimisation of high-rise buildings has been the focus of researchers because of significant performance enhancement, mainly in energy consumption and generation. However, optimisation of self-sufficient high-rise buildings requires the integration of multiple performance aspects related to the vital resources of human beings (e.g., energy, food, and water) and consideration of large numbers of design parameters related to these multiple performance aspects. Hence, the complexity of self-sufficient high-rise buildings is more challenging than optimising regular high-rises that have not been addressed in the literature. The purpose of this dissertation is to present a framework for performance optimisation of self-sufficient high-rise buildings using artificial intelligence focusing on the conceptual phase of the design process.

Chapter 1 is the introduction to the dissertation. The necessity and the definition of the self-sufficient high-rise buildings are explained after presenting recent proposals of scholars and professionals related to sustainable living alternatives. Additionally, the complexity level of self-sufficiency, which consists of four categories as scale, period, parameters, and performance, is described by indicating the focus in the overall chart. Until now, high-rise buildings have been optimised to improve the energy performance that reflects self-sufficiency only in energy consumption. The contribution of this study, which focuses on optimising high-rise buildings for multiple resources (e.g., energy, food, and water) to decrease their environmental impact, is described. The research method consists of four main steps: literature review, tool development and pilot study, computational method development, and case study. After presenting the research problem, questions, aim, objectives, and output of the dissertation, the research method explains the abovementioned steps. Finally, the chapter is concluded by discussing the social and scientific relevance of the research.

Chapter 2 presents the literature review on optimising form-finding parameters in performative computational architecture that entails form generation, performance evaluation, and optimisation. A systematic review is conducted based on multiple databases to elaborate the trends for investigating well-performing design alternatives using optimisation algorithms in the architectural design domain. Therefore, the review focuses on studies involving form-finding parameters. One hundred studies are systematically reviewed, focusing on swarm and evolutionary optimisation algorithms frequently used in architectural design. The chapter concludes by presenting the gaps and needs considered while developing the optimisation tool and computational framework, focusing on form-finding parameters, performance evaluation, and optimisation applications.

Chapter 3 presents the development of the optimisation tool called Optimus and the pilot study to test the efficiency of the multi-zone optimisation approach in high-rises. Part A of Chapter 3 presents the Optimus tool, which considers a self-adaptive ensemble evolutionary algorithm that can cope with large numbers of design parameters. Tests 1 and 2 are presented to indicate the relevance of the developed tool based on 30-dimensional Congress on Evolutionary Computation 2005 benchmark problems and a 70-dimensional design problem. Part B explains Test 3 to utilise the efficiency of the multi-zone optimisation approach. The main idea of this method is to divide the building into several subdivisions (zones) to be considered different optimisation problems. The pilot high-rise model considers one of the most used façade parameters reported in Chapter 2 (overhang length) and glazing type for two conflicting daylight metrics predicted by the basic version of artificial neural network models and optimised by the initial version of Optimus tool.

Chapter 4 presents the multi-zone optimisation (MUZO) methodology that entails the parametric high-rise model, machine learning for surrogate models, computational optimisation, and decision-making. Part A of this chapter presents the entire methodology and two design scenarios indicated as Tests 4 and 5 to demonstrate the relevance of the MUZO. Both scenarios, focusing on quad-grid and diagrid façade designs, integrate frequently used form-finding parameters for building shape and façade design reported in Chapter 2. Additionally, Part A conducts the machine learning results using the parametric high-rise models to cope with the computationally expensive simulation time while assessing the performance of the entire building. Afterwards, Part B presents the optimisation problems and results of both design scenarios using the predictive models developed in Part A and the released version of the Optimus tool presented in Chapter 3. Since the study focuses on optimising the entire design of the high-rise scenarios are considered 260 and 220 design parameters, respectively, for quad-grid and diagrid scenarios. Consequently, Part B presents the relevance of the MUZO methodology by comparing the results with the regular high-rise scenarios, which use the same design parameters in the entire building.

Chapter 5 investigates utilising the MUZO methodology and Optimus tool to optimise the Europoint complex in Rotterdam, the Netherlands, for self-sufficiency in terms of energy consumption and food production. The sufficiency in food production is demonstrated for lettuce crops grown in vertical farms. Building-integrated photovoltaic panels are used in several building parts regarding sufficiency in energy. The optimisation problem, which involves 117 decision variables related to the façade design, and the thermal properties of the glazing, addresses the self-sufficiency at the building scale in detail. Moreover, another optimisation problem reports the potentials at the neighbourhood scale using the same self-sufficiency aspects and design parameters. Among 13 algorithms used to optimise both problems, the Optimus tool presented the most favourable self-sufficiency performance.

Chapter 6 concludes the dissertation by summarising the contribution of the research, addressing the answers to research questions, presenting the limitations of the research, and highlighting future recommendations. After completing the development of the optimisation tool and conducting preliminary results of the pilot high-rise model, the research results are conducted in weeks instead of years during the development of the MUZO methodology and case study. Thanks to artificial intelligence, decision-makers can utilise the proposed computational framework for optimising self-sufficient high-rise buildings. In this way, consequences of the decisions on performance aspects of self-sufficiency become possible for such a complex design task with high awareness of the alternatives in search space within a reasonable timeframe.

Samenvatting

Bevolkingsgroei en verstedelijkingstrends hebben veel gevolgen voor de toename van het wereldwijde energieverbruik, de CO2-uitstoot en de afname van het bouwland per persoon. Alternatieve ontwerpvoorstellen voor duurzaam wonen staan op de agenda van onderzoekers en professionals om in te spelen op de behoeften van de 21e eeuw voor een duurzame toekomst. Sinds de vroege voorbeelden in de 19e eeuw is hoogbouw een van de onvermijdelijke typologieën van metropolen om extra vloeroppervlak te bieden in compacte steden. Gebaseerd op de feiten van de 21e eeuw, zou hoogbouw meer moeten vervullen dan extra vloeroppervlak bieden op het beperkte stedelijke perceel. Dit onderzoek suggereert "zelfvoorzienende hoogbouw" die naast dichte bewoning voor duurzaam leven, essentiële hulpbronnen kan genereren en efficiënt verbruiken. Optimalisatie van hoogbouw is de focus van onderzoekers geweest vanwege aanzienlijke prestatieverbetering, voornamelijk in energieverbruik en opwekking. Optimalisatie van zelfvoorzienende hoogbouw vereist echter de integratie van meerdere prestatie-aspecten die verband houden met de vitale hulpbronnen van de mens (bijv. energie, voedsel en water) en het in overweging nemen van een groot aantal ontwerpparameters met betrekking tot deze meerdere prestatie-aspecten. Daarom is de complexiteit van zelfvoorzienende hoogbouw een grotere uitdaging dan het optimaliseren van reguliere hoogbouw die niet in de literatuur is behandeld. Het doel van dit proefschrift is om een raamwerk te presenteren voor prestatie-optimalisatie van zelfvoorzienende hoogbouw met behulp van kunstmatige intelligentie, gericht op de conceptuele fase van het ontwerpproces.

Hoofdstuk 1 is de inleiding tot het proefschrift. De noodzaak en de definitie van de zelfvoorzienende hoogbouw worden uitgelegd na presentatie van recente voorstellen van wetenschappers en professionals met betrekking tot duurzame woonalternatieven. Bovendien wordt het complexiteitsniveau van zelfvoorziening, dat uit vier categorieën bestaat als schaal, periode, parameters en prestatie, beschreven door de focus in de algemene grafiek aan te geven. Tot nu toe is hoogbouw geoptimaliseerd om de energieprestaties te verbeteren, die alleen op het gebied van energieverbruik een weerspiegeling zijn van zelfvoorziening. De bijdrage van deze studie, die zich richt op het optimaliseren van hoogbouw voor meerdere bronnen (bijvoorbeeld energie, voedsel en water) om hun impact op het milieu te verminderen, wordt beschreven. De onderzoeksmethode bestaat uit vier hoofdstappen: literatuuronderzoek, tool ontwikkeling en pilotstudie, ontwikkeling van computationele methoden en case study. Na het presenteren van het onderzoeksprobleem, de vragen, het doel, de doelstellingen en de output van het proefschrift, legt de onderzoeksmethode de bovengenoemde stappen uit. Ten slotte wordt het hoofdstuk afgesloten met een bespreking van de maatschappelijke en wetenschappelijke relevantie van het onderzoek.

Hoofdstuk 2 presenteert de literatuurstudie over het optimaliseren van parameters voor het vinden van formulieren in performatieve computationele architectuur, waarbij vormen worden gegenereerd, prestatie-evaluatie en optimalisatie worden uitgevoerd. Er wordt een systematische review uitgevoerd op basis van meerdere databases om de trends uit te werken voor het onderzoeken van goed presterende ontwerpalternatieven met behulp van optimalisatie-algoritmen in het architectonisch ontwerpdomein. Daarom richt de review zich op studies met parameters voor het vinden van vormen. Honderd onderzoeken worden systematisch beoordeeld, gericht op zwerm- en evolutionaire optimalisatie-algoritmen die vaak worden gebruikt in architectonisch ontwerp. Het hoofdstuk besluit met een presentatie van de hiaten en behoeften waarmee rekening is gehouden bij het ontwikkelen van de optimalisatietool en het rekenraamwerk, waarbij de nadruk ligt op parameters voor het vinden van vormen, prestatie-evaluatie en optimalisatie toepassingen.

Hoofdstuk 3 presenteert de ontwikkeling van de optimalisatietool Optimus en de pilotstudie om de efficiëntie van de multi-zone optimalisatiebenadering in hoogbouw te testen. Deel A van hoofdstuk 3 presenteert de Optimus-tool, die een zelf-aanpassend evolutionair ensemble-algoritme beschouwt dat een groot aantal ontwerpparameters aan kan. Tests 1 en 2 worden gepresenteerd om de relevantie van de ontwikkelde tool aan te geven op basis van de benchmark problemen van het 30-dimensional Congress on Evolutionary Computation 2005 (CEC2005 benchmark problemen) en een 70-dimensionaal ontwerpprobleem. Deel B legt Test 3 uit om de efficiëntie van de multi-zone optimalisatiebenadering te benutten. Het belangrijkste idee van deze methode is om het gebouw op te delen in verschillende onderverdelingen (zones) om als verschillende optimalisatieproblemen te worden beschouwd. Het pilootmodel voor hoogbouw houdt rekening met een van de meest gebruikte gevel parameters zoals gerapporteerd in hoofdstuk 2 (overhanglengte) en het type beglazing voor twee tegenstrijdige daglichtstatistieken, voorspeld door de basisversie van kunstmatige neurale netwerkmodellen en geoptimaliseerd door de initiële versie van de Optimus-tool.

Hoofdstuk 4 presenteert de Multi-Zone Optimalisatie (MUZO) methodologie die het parametrische hoogbouwmodel, Machine Learning voor Surrogate modellen, Computationele optimalisatie en besluitvorming omvat. Deel A van dit hoofdstuk presenteert de gehele methodologie en twee ontwerpscenario’s aangeduid als

Tests 4 en 5 om de relevantie van de MUZO aan te tonen. Beide scenario's, gericht op quad-grid- en diagrid-gevel ontwerpen, integreren veelgebruikte parameters voor het vinden van vormen voor de vorm van gebouwen en het gevelontwerp zoals beschreven in hoofdstuk 2. Bovendien voert deel A de machine learning-resultaten uit, met behulp van de parametrische hoogbouw modellen, om het hoofd te bieden aan de rekenkundig dure simulatietijd bij het beoordelen van de prestaties van het hele gebouw. Daarna presenteert deel B de optimalisatie problemen en resultaten van beide ontwerpscenario's met behulp van de voorspellende modellen die zijn ontwikkeld in deel A en de vrijgegeven versie van de Optimus-tool gepresenteerd in hoofdstuk 3. Aangezien de studie zich richt op het optimaliseren van het gehele ontwerp van de hoogbouwscenario's, worden 260 en 220 ontwerp parameters voor respectievelijk quad-grid- en diagrid-scenario's. Deel B presenteert daarom de relevantie van de MUZO-methodiek door de resultaten te vergelijken met de reguliere hoogbouwscenario's, die dezelfde ontwerpparameters in het hele gebouw gebruiken.

Hoofdstuk 5 onderzoekt het gebruik van de MUZO-methodologie en Optimus-tool om het Europoint-complex in Rotterdam-Nederland te optimaliseren voor zelfvoorziening in termen van energieverbruik en voedselproductie. De toereikendheid van de voedselproductie is aangetoond voor slagewassen die worden geteeld in verticale boerderijen. Gebouw geïntegreerde fotovoltaïsche panelen worden gebruikt in verschillende delen van gebouwen met betrekking tot voldoende energie. Het optimalisatieprobleem, dat betrekking heeft op 117 beslissingsvariabelen met betrekking tot het gevelontwerp en de thermische eigenschappen van de beglazingen, gaat in detail in op de zelfvoorziening op gebouwschaal. Bovendien rapporteert een ander optimalisatie probleem de potentiëlen op buurtschaal met dezelfde zelfvoorzieningsaspecten en ontwerpparameters. Van de 13 algoritmen die werden gebruikt om beide problemen te optimaliseren, vertoonde de Optimus-tool de meest gunstige zelfvoorzieningsprestaties.

Hoofdstuk 6 besluit het proefschrift door de bijdrage van het onderzoek samen te vatten, de antwoorden op onderzoeksvragen te behandelen, de beperkingen van het onderzoek te presenteren en toekomstige aanbevelingen te benadrukken. Na afronding van de ontwikkeling van de optimalisatietool en het uitvoeren van voorlopige resultaten van het pilot-hoogbouwmodel, worden de onderzoeksresultaten in weken in plaats van jaren uitgevoerd tijdens de ontwikkeling van de MUZO-methodiek en de case study. Dankzij kunstmatige intelligentie kunnen besluitvormers het voorgestelde rekenraamwerk gebruiken om zelfvoorzienende hoogbouw te optimaliseren. Op deze manier worden consequenties van de beslissingen over prestatieaspecten van zelfvoorziening mogelijk voor zo'n complexe ontwerptaak, met een hoog bewustzijn van de alternatieven in de zoekruimte binnen een redelijk tijdsbestek.

INTRODUCTION | METHOD(S) | CHALLENGE(S)

LITERATURE REVIEW

	METHOD(S)	CHALLENGE(S)	
J1 **RQ1**	Optimising form-finding parameters in performative computational architecture	■ 17 Form-finding parameters ● 13 Performance aspects ▲ 12 Evolutionary algorithms 3 Swarm algorithms	

TOOL DEVELOPMENT AND PILOT STUDY

	METHOD(S)	CHALLENGE(S)	
J2 **RQ2**	PART A Developing Optimus tool using self-adaptive ensemble evolutionary algorithm	■ 30 Parameters ▲ 4 Optimisation algorithms	TEST 1
		■ 70 Parameters ▲ 4 Optimisation algorithms	TEST 2
J3 **RQ3**	PART B Preliminary results of multi-zone approach using pilot high-rise model	■ 100 Parameters ● 2 Daylight metrics ▲ 5 ANN models 1 Optimisation algorithm	TEST 3

METHODOLOGICAL FRAMEWORK

	METHOD(S)	CHALLENGE(S)	
J4 **RQ4**	PART A Introducing multi-zone optimisation (MUZO) methodology and prediction results of quad-grid and diagrid high-rise scenarios	■ 260 Parameters ● 2 Daylight metrics ▲ 20 ANN models	TEST 4
		■ 220 Parameters ● 2 Daylight metrics ▲ 20 ANN models	TEST 5
J5 **RQ5**	PART B Optimising high-rise scenarios using the predictive models with Optimus and validation of the MUZO methodology	■ 260 Parameters ● 20 Predictive models ▲ 3 Optimisation algorithms	TEST 4
		■ 220 Parameters ● 20 Predictive models ▲ 3 Optimisation algorithms	TEST 5

CASE STUDY

	METHOD(S)	CHALLENGE(S)
J6 **RQ6**	Optimising Europoint complex for self-sufficiency in energy consumption and food production using MUZO and Optimus	■ 117 Parameters ● 1 Self-sufficiency in energy 1 Self-sufficiency in food 1 Daylight metric ▲ 45 ANN models 13 Optimisation algorithms

CONCLUSIONS

J Journal
RQ Research question

■ Parameter ● Performance ▲ AI Method

TOC

1 General introduction

1.1 Background

The world deals with population growth and urbanisation trends. According to a recent report [1], United Nations foresees that by 2050, 2.5 billion more people will be living in the world. In three decades, 68% of the global population is expected to live in urban areas instead of 55% in 2018 and 30% in 1950. The report also projects an increase in megacities, which inhabit more than 10 million people, from 33 to 43 in a decade. One of the main consequences of population growth and urbanisation trends is an increasing demand for high-rise buildings in metropolises [2]. The researchers have focussed on performance optimisation of high-rises because of significant energy consumption compared to low rise buildings. Therefore, one may argue that the ultimate purpose of recent high-rise optimisation studies is to decrease the total energy consumption. However, the consequences of population growth and urbanisation trends are not limited to energy usage.

The United Nations Food and Agriculture Organisation foresees that only one-third of the arable land per person in 1970 will be available in 2050 [3]. Therefore, scholars and professionals focus on alternative living proposals for a sustainable future. Kirimtat et al. [4-8] investigated smart floating cities and floating neighbourhoods to provide extra floor space for habitation and agriculture considering renewable resources. Even though floating settlements are potentially sustainable alternatives to cope with population growth, they do not address the challenges in metropolises. Therefore, the next generation of high-rise buildings must provide more than extra floor space in the limited urban plot. Within the Paris Smart City 2050 concept, Vincent Callebaut Architectures proposed eight prototypes of positive energy towers [9]. The projects aim to deal with CO_2 emissions and the urban heat island effect while providing renewable and recyclable energy for the buildings and surroundings.

On the other hand, Singapore is one of the leading cities with many examples of green architecture since the layers of plants can temper the heat, filter pollutants, and absorb rainwater [10]. Additionally, a recent example called Vertical Forest in Milan, designed by Boeri Studio, suggests an alternative living for human beings with other species considering layers of plants in various floor levels [11]. These projects present promising sustainable alternatives in metropoles. Nevertheless, the performance optimisation of the self-sufficient high-rise buildings has not been addressed during the conceptual design phase.

As the next generation of high-rises, the definition of self-sufficient high-rise buildings is identified as those that combine dense habitation, generation, and efficient use of resources (e.g., energy, food, water) in one building. Therefore, self-sufficient high-rise buildings are different from the studies related to only self-sufficiency in energy (e.g., net-zero energy buildings, energy-autonomous buildings [12]). Additionally, the current technology limits constructing self-sufficient high-rise buildings as off-grid systems, e.g. small-scaled autonomous houses [13]. Therefore, the contribution of self-sufficient high-rises can be generating and efficiently using multiple resources (such as energy, food, water), decreasing their environmental impact, and providing dense habitation with mixed-used building programs in metropolitan areas. This suggests a new design problem in architecture that has multiple challenges to cope with in the conceptual phase of the architectural design process.

High-rise buildings are one of the most complex tasks in the architecture discipline. Because various design parameters (e.g., building form, façade design), which affect the performance of the building, are involved in the design process. Since the decisions given in the conceptual phase affect the buildings' overall performance, optimisation methods have been widely used to investigate well-performing high-rise design alternatives. However, the complexity of the self-sufficient high-rise design problem is more challenging than optimising high-rise buildings only for energy or daylight. The main reason is that optimising high-rises for self-sufficiency necessitates providing sufficient resources for residents in addition to efficient usage. Instead of focusing on a specific floor or part of the building, high-rises in dense urban areas also require optimising the entire design of the building because of performance variances between ground and sky floor levels. Therefore, the overall design of the high-rise building should be addressed to achieve high performance in self-sufficiency. Briefly, the complexity of the self-sufficient high-rise design problem is related to 1) providing sufficient resources; 2) coping with the parameters of the building; 3) integrating multiple performance aspects.

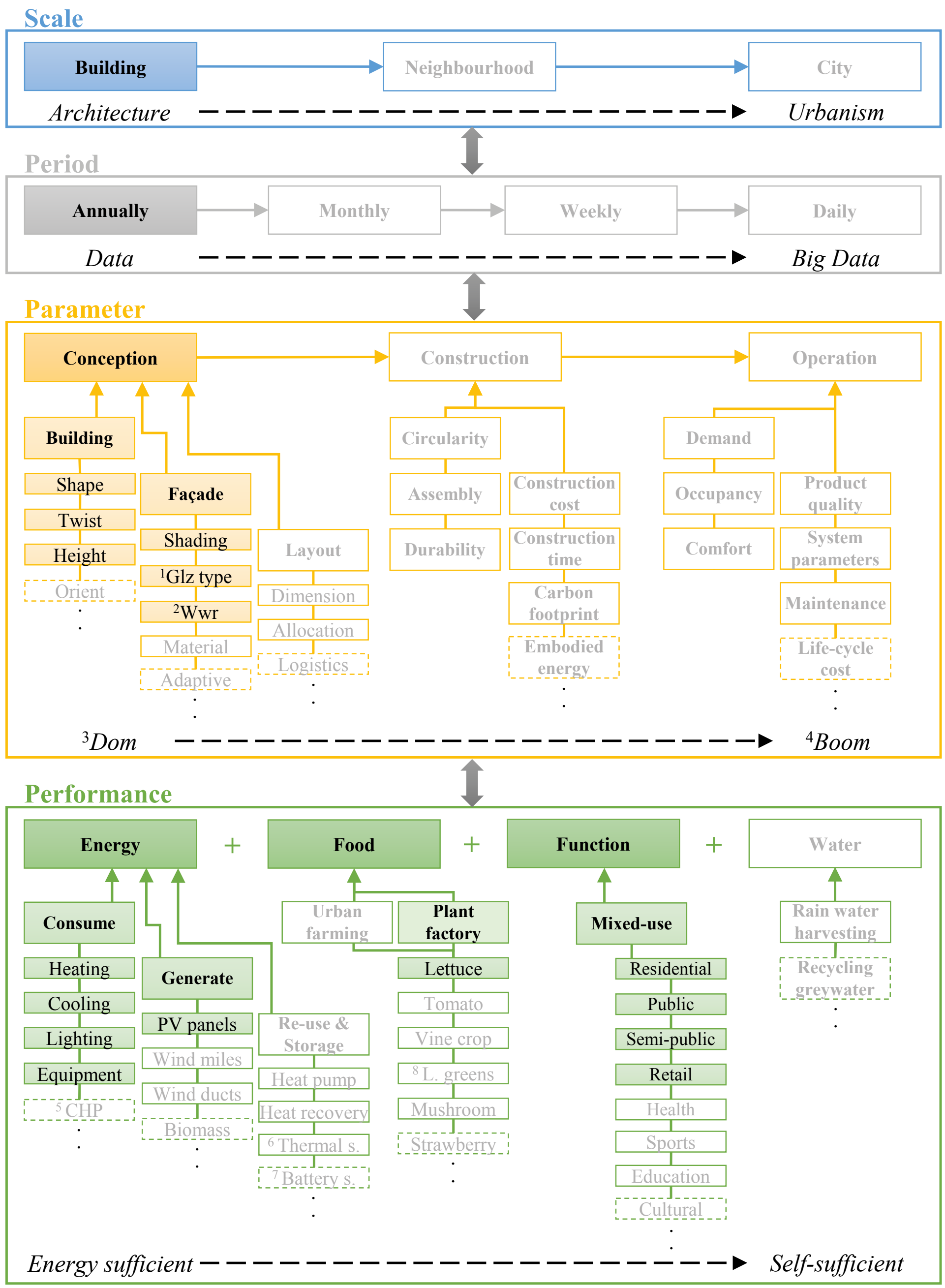
Providing sufficient resources is related to the sufficiency scale and period. The larger scale to provide resources, such as providing lettuce crops for the residents of the high-rise building and the people who live in the neighbourhood, suggests a more challenging task than providing resources only for the high-rise residents. Regarding the period of self-sufficiency, focusing on an annually self-sufficient building is less challenging than monthly or weekly self-sufficient building. Optimisation of the annual self-sufficiency in a high-rise design reflects the consideration of the overall performance of the building in one year. On the other hand, designing a high-rise building that is expected to be monthly self-sufficient brings along providing self-sufficiency in each month. As a result, the monthly performance of the building, which suggests an additional challenge, should be involved during the optimisation process.

Coping with the parameters of the self-sufficient building is another reason for the complexity. During the conceptual phase of the design, decisions on parameters usually affect the buildings' functioning and performance. Decision making for the entire design while considering the specific part of the high-rise building suggests a limited approach in metropolises owing to the built environment's density. Therefore, self-sufficient high-rises should consider the suitable decisions for various building parts to achieve high performance in the focussed criteria. The design of the high-rise building, as many other buildings, are related to the shape, façade, and layout parameters. Different decisions for various building parts multiply the design complexity due to each part's unique parameter selection. Finally, decisions on additional parameters are necessary while realising the project during the construction phase and operating the building.

Integrating multiple performance aspects also increase the complexity of the problem. Even though optimising a performance aspect to maximise the advantages and minimise the disadvantages of the design solution is already a challenging task, considering multiple aspects suggests an additional complexity. The first reason is that the design decision should simultaneously reflect desirable performance results for all the aspects. In most cases, performance aspects conflict with each other (such as optimising energy consumption and daylight), that increase the complexity of the design problem for finding an optimal solution for both aspects. Secondly, simulation time of the performance aspects suggests an additional challenge because the aspects related to the self-sufficient high-rises mainly involve the simulations. Optimising a design problem considering a simulation-based performance evaluation takes a significant time. Optimisation of the entire high-rise design for multiple simulation-based performance aspects requires an enormous computation time, in most cases years, to complete the optimisation process.

Four main categories, which presents the overall framework of the self-sufficient high-rise buildings, are defined as in Fig. 1.1. The first two categories indicate the scale and the period of providing sufficient resources. The third category presents the parameters of different phases starting from conception until the operation of the building. The final category suggests the performance aspects to be considered for generation and efficient usage in the self-sufficient high-rise design. In addition to the resources (energy, food, water), the function of the high-rise building is involved as another performance aspect. Because a mixed-use building program can reduce CO_2 emissions and improve the life quality of the building's residents as most of the demands can be fulfilled in the same living environment.

This research focuses on annually self-sufficient high-rise buildings on a building scale while dealing with the decisions in the conceptual phase considering energy, food, and multiple building functions. The development of an optimisation tool (Optimus) and proposal of an optimisation method (multi-zone optimisation) is in the scope of this dissertation to cope with the complexity of the design problem. Artificial intelligence, which involves machine learning and optimisation in this research, supports the decision making to predict and optimise the performance of the entire building while considering large numbers of parameters. The research output addresses architects and engineers who specialise in sustainable design solutions, high-rise buildings, building performance simulation, and artificial intelligence in architectural design problems.



[1] Glazing type [2] Window-to-wall ratio [3] Design optimisation model [4] Building operation optimisation Model [5] Combined heat and power [6] Thermal storage [7] Battery systems [8] Leafy greens

Focus of this work

FIG. 1.1 Framework of self-sufficient high-rise buildings

TOC

1.2 Problem statement

The current state of the art focuses on optimising high-rise buildings for efficient usage of resources, such as natural ventilation potential [14], efficient energy usage with façade design [15], optimising building operation scenarios [16], and layout plan for energy-efficiency [17]. Regarding the generation of resources, limited studies can be found related to optimising high-rise buildings in the conceptual design phase [18-21]. Although related works present promising contributions to decrease the energy consumption of high-rises, these are only limited to the energy aspect. None of the challenges mentioned in the complexity of self-sufficient high-rise buildings is dealt with in the literature. Therefore, the development of a computational framework, which can cope with the optimisation of the entire high-rise design for self-sufficiency in multiple resources, is required.

1.3 Research questions, aim, and objectives

Research questions

In the light of the general background and the statement of the problem explained in previous sections, this research addresses the main research question as follows:

How can we optimise the performance of self-sufficient high-rise buildings using artificial intelligence in the conceptual design phase?

To answer the main research question, six sub-questions, which are addressed in different chapters of the dissertation, are defined as follows:

- What is the state-of-the-art for optimising form-finding parameters using swarm and evolutionary algorithms in performative computational architecture (Chapter 2)?

- What kind of algorithms can discover promising performance results for design problems with large numbers of form-finding parameters in the architecture domain (Chapter 3A)?

- Can the multi-zone optimisation approach provide better performance results for high-rise buildings in dense urban environments (Chapter 3B)?

- How can we reach swift and accurate predictions for computationally expensive performance aspects of sustainable buildings in the entire design of high-rises (Chapter 4A)?

- How can we optimise large numbers of design parameters while investigating multiple performance aspects of sustainable high-rise alternatives (Chapter 4B)?

- What is the potential of the developed computational method for self-sufficiency in energy consumption and food production at the building and neighbourhood scales (Chapter 5)?

Research Aim

The research aims to develop a computational framework to optimise self-sufficient high-rise buildings during the conceptual design phase. The goal is to allow architects and engineers to make design decisions in a reasonable timeframe with high awareness of the building performance considering multiple self-sufficiency aspects while coping with large numbers of design parameters.

Research Objectives

Concerning the aim of the research, the following objectives are defined:

- Investigating the state-of-the-art on optimising form-finding parameters in the domain of performative computational architecture. (Chapter 2)

- Developing an optimisation tool for design problems having large numbers of parameters in the architecture domain. (Chapter 3A)

- Conducting preliminary results of multi-zone optimisation approach in a highly dense built environment. (Chapter 3B)

- Developing a computational framework for optimising multiple floor levels while coping with computationally expensive simulations and optimising large numbers of design parameters. (Chapter 4)

- Demonstrating the developed computational framework for potential self-sufficiency at the building and neighbourhood scales. (Chapter 5)

1.4 Research output

This research produces a computational framework to optimise high-rise buildings for self-sufficiency performance aspects in the conceptual phase of the design process. The developed method, called multi-zone optimisation (MUZO), consists of three phases: parametric high-rise model, machine learning for surrogate models, computational optimisation and decision-making. Each phase of the MUZO consists of several computational workflows that allow swift performance evaluation of self-sufficient high-rise buildings with high awareness of the search space. Since optimising large numbers of design parameters remains a challenge, the research also produces an optimisation tool, based on collaborative work, called Optimus, to be used in the optimisation phase of MUZO. Consequently, self-sufficiency in energy consumption and food production (for lettuce crops) are demonstrated for Europoint complex in Rotterdam, the Netherlands, using MUZO methodology with Optimus tool.

1.5 Research method

The research considers mixed methods based on quantitative analyses, consisting of four main steps to answer the research questions addressed in the previous section (Fig. 1.2). In the first step – Chapter 2, the literature review presents the concept called performative computational architecture and relevant information in form-finding parameters, performance aspects, and optimisation algorithms. In the second step – Chapter 3, Part A develops the optimisation tool by validating its efficiency using benchmark problems (Test 1) and a design problem (Test 2). Part B presents the preliminary results of the MUZO methodology using the developed optimisation tool (Test 3). In the third step – Chapter 4, Part A presents all the phases of the MUZO methodology in detail and develops advanced machine learning models for two high-rise scenarios (Tests 4 and 5). Part B of the third step uses the developed machine learning models in Part A to validate the relevance of the MUZO methodology by optimising two high-rise scenarios using the developed optimisation tool (Tests 4 and 5). The last step focuses on the Europoint complex as a case study to utilise the MUZO methodology and the developed optimisation tool. Actions taken in each step are explained below.

FIG. 1.2 Overview of the research method

Step 1 (Ch. 2): Literature Review

Studies focusing on optimising form-finding parameters are systematically investigated within the scope of performative computational architecture. The scope is limited to swarm and evolutionary optimisation algorithms since they are mostly used in this domain. Findings on parameter usage and application of optimisation algorithm are used in the next steps of the research method. The chapter is published as:

[22] Ekici, B.; Cubukcuoglu, C.; Turrin, M.; Sariyildiz, I. S., Performative computational architecture using swarm and evolutionary optimisation: A review. Building and environment 2019, 147, 356-371.

Step 2 (Ch. 3): Tool Development and Pilot Study (Tests 1, 2, and 3)

Part A of step two focuses on developing an optimisation tool called Optimus, which works as a plug-in in Grasshopper 3d algorithmic modelling environment. The developed tool is tested with 20 benchmark problems (CEC 2005 problems), which are frequently used in the domain of evolutionary computation, with 30 decision variables (Test 1); and a 70-dimensional design problem (Test 2). The results of three other algorithms are compared with the results of Optimus.

Part B of this chapter investigates the performance enhancement of the MUZO approach with 100 decision variables in five building subdivisions considering two daylight metrics, basic predictive models, overhang length (one of the most used parameters in the reviewed papers) and glazing type. The results, conducted by the initial version of the Optimus tool, are compared with the initial and regular design scenarios (Test 3). The chapter is published as:

[23] Cubukcuoglu, C.; Ekici, B.; Tasgetiren, M. F.; Sariyildiz, S., OPTIMUS: self-adaptive differential evolution with ensemble of mutation strategies for grasshopper algorithmic modeling. Algorithms 2019, 12, (7), 141.

[24] Ekici, B.; Kazanasmaz, T.; Turrin, M.; Tasgetiren, M. F.; Sariyildiz, I. S. A Methodology for daylight optimisation of high-rise buildings in the dense urban district using overhang length and glazing type variables with surrogate modelling, Journal of Physics: Conference Series, 2019; IOP Publishing: 2019; p 012133.

Step 3 (Ch. 4): Methodological Framework (Tests 4 and 5)

Part A of step three introduces the phases of MUZO, which are parametric high-rise model, machine learning for surrogate models, computational optimisation and decision-making. Two high-rise scenarios, which consider form-finding parameters and thermal properties of the glazing, are investigated. The first scenario has a quad-grid façade design with 260 design parameters (Test 4). The other scenario has a diagrid façade with 220 dimensions (Test 5). The form-finding parameters are selected among the most used building shape and façade design parameters reported in the literature review. Additionally, thermal properties of the glazing materials are involved after promising performance improvement was observed in the pilot study. Results of 40 predictive models are conducted using advanced machine learning applications developed in Python, and prediction accuracies are compared with similar studies in the same domain.

Part B of step three validates the relevance of the MUZO methodology by comparing the optimised design alternatives with regular high-rise scenarios. During the optimisation process, predictive models developed for quad-grid and diagrid scenarios (Tests 4 and 5) are also used in this part. The optimisation process involves the released version of the Optimus tool in addition to two other well-known algorithms in Grasshopper 3d. The chapter is published as:

[25] Ekici, B.; Kazanasmaz, Z. T.; Turrin, M.; Taşgetiren, M. F.; Sariyildiz, I. S., Multi-zone optimisation of high-rise buildings using artificial intelligence for sustainable metropolises. Part 1: Background, methodology, setup, and machine learning results. Solar Energy 2021, 224, 373-389.

[26] Ekici, B.; Kazanasmaz, Z. T.; Turrin, M.; Taşgetiren, M. F.; Sariyildiz, I. S., Multi-zone optimisation of high-rise buildings using artificial intelligence for sustainable metropolises. Part 2: Optimisation problems, algorithms, results, and method validation. Solar Energy 2021, 224, 309-326.

Step 4 (Ch. 5): Case Study

The last step of the research method focuses on optimising the Europoint complex in Rotterdam, the Netherlands, for self-sufficiency in energy consumption and food production using the MUZO methodology. As the first phase of MUZO, a parametric high-rise model is developed for three towers of the complex involving vertical farms for lettuce crops and building integrated photovoltaic panels for energy generation. The second phase of MUZO predicts the performance of self-sufficiency using 45 surrogate models. In the final phase, an optimisation problem that consists of 117 decision variables, two objective functions for self-sufficiency in energy and food, and one constraint function for daylight availability is introduced. Thirteen optimisation algorithms, including the Optimus, are employed to investigate the self-sufficiency using two problem formulations. While the first formulation considers a single-objective constrained problem for the building scale, the second formulation focuses on a multi-objective constrained problem for the neighbourhood scale. A deep investigation is conducted at the building scale, which provides a high awareness of the search space using the entire optimisation framework of MUZO (algorithm comparison and replication). Finally, potential self-sufficiency at the neighbourhood scale is presented. The chapter is published as:

[27] Ekici, B.; Turkcan, O. F. S. F.; Turrin, M.; Sariyildiz, I. S.; Tasgetiren, M. F., Optimising High-Rise Buildings for Self-Sufficiency in Energy Consumption and Food Production Using Artificial Intelligence: Case of Europoint Complex in Rotterdam. Energies 2022, 15, (2), 660.

1.6 Social and scientific relevance

Optimisation of high-rise buildings is a complex task. In the case of high-rise optimisation for self-sufficiency performance aspects, the complexity increases due to the additional challenges explained in the background section. Therefore, the method to cope with this complex problem should provide well-performing design alternatives with respect to large numbers of design parameters and complete the design investigation involving multiple performance aspects in a reasonable time. Accordingly, the relevance of this research has two folds as scientific and societal.

The scientific relevance of the thesis is expanding the knowledge on optimising complex high-rise design problems. The proposed novel computational framework can optimise multiple performance aspects and cope with large numbers of design parameters. Moreover, the multi-zone optimisation approach allows optimising the entire high-rise design considering the impact of the surrounding in dense urban environments. The research also proves that the performance of optimisation algorithms differs according to the nature of the design problems. Therefore, the thesis also contributes to approaching optimisation problems in the architectural design domain, considering comparisons of different algorithms. Finally, the relevance of the developed optimisation tool for the architectural design domain, regarding the ability to optimise large numbers of parameters, is validated during the different steps of the research.

The societal relevance of the thesis is to support designers, architects, and engineers by providing self-sufficient high-rise alternatives in a reasonable time during the conceptual phase of the design process. Because most of the performance aspects of self-sufficiency require simulation, investigating well-performing alternatives with high awareness of the search space requires significant time. Thanks to artificial intelligence involved in the proposed computational framework, decision-makers can identify self-sufficient design solutions with high awareness of the consequences of design alternatives in the search space in weeks instead of decades. Therefore, the research also supports professionals for designing self-sufficient high-rise buildings as alternative sustainable living proposals. In this way, high-rises' environmental impact can be decreased while fulfilling vital resources of human beings in metropolises.

References

[1] UN, World Urbanization Prospects: The 2018 Revision (ST/ESA/SER. A/420). *United Nations, Department of Economic and Social Affairs, Population Division: New York, NY, USA* **2019**.

[2] CTBUH, The Skyscraper Center, Council on Tall Buildings and Urban Habitat, http://www.skyscrapercenter.com (accessed on: 21.02.2022). **2020**.

[3] Food and Agriculture Organization of the United Nations (FAO). “Database on Arable Land.” http://data.worldbank.org/indicator/AG.LND.ARBL.HA.PC?end%20&hx003D;2013&hx0026;start%20&hx003D;1961&hx0026;view&hx003D;chart (accessed on: 10.11.2021).

[4] Kirimtat, A.; Chatzikonstantinou, I.; Sariyildiz, S.; Tartar, A. In *Designing self-sufficient floating neighborhoods using computational decision support*, IEEE Congress on Evolutionary Computation (CEC), **2015**; pp 2261-2268.

[5] Kirimtat, A.; Ekici, B.; Cubukcuoglu, C.; Sariyildiz, S.; Tasgetiren, F., Evolutionary algorithms for designing self-sufficient floating neighborhoods. In *Optimization in Industry*, Springer: **2019**; pp 121-147.

[6] Kirimtat, A.; Krejcar, O. In *Development of self-sufficient floating cities with renewable resources*, International Conference on Computational Collective Intelligence, Springer: **2018**; pp 437-446.

[7] Kirimtat, A.; Krejcar, O.; Tasgetiren, M. F. In *Evolutionary Computation for the Development of Smart Floating Cities*, IEEE International Conference on Industrial Technology (ICIT), **2020**; pp 822-828.

[8] Kirimtat, A.; Krejcar, O.; Tasgetiren, M. F.; Herrera-Viedma, E., Multi-performance based computational model for the cuboid open traveling salesman problem in a smart floating city. *Building and Environment* **2021,** 196, 107721.

[9] Vincent Callebaut Architects, Paris Smart City 2050 https://vincent.callebaut.org/object/150105_parissmartcity2050/parissmartcity2050/projects (accessed on: 21.02.2022).

[10] https://www.todayonline.com/singapore/singapore-leads-green-architecture-sustainable-building-design-asia (accessed on: 21.12.2022).

[11] Boeri Studio, Vertical Forest, https://www.stefanoboeriarchitetti.net/en/project/vertical-forest/ (accessed on: 21.02.2022).

[12] Voss, K.; Musall, E.; Lichtmeß, M., From low-energy to net zero-energy buildings: status and perspectives. *Journal of Green building* **2011,** 6, (1), 46-57.

[13] Vale, B.; Vale, R., The new autonomous house: Design and planning for sustainability. **2000**.

[14] Li, Y.; Li, X. In *Natural ventilation potential of high-rise residential buildings in northern China using coupling thermal and airflow simulations*, Building Simulation, Springer: **2015**; pp 51-64.

[15] Wang, M.; Hou, J.; Hu, Z.; He, W.; Yu, H. In *Optimisation of the double skin facade in hot and humid climates through altering the design parameter combinations*, Building Simulation, Springer: **2021**; pp 511-521.

[16] Chen, X.; Yang, H., A multi-stage optimization of passively designed high-rise residential buildings in multiple building operation scenarios. *Applied energy* **2017,** 206, 541-557.

[17] Gan, V. J.; Wong, H.; Tse, K. T.; Cheng, J. C.; Lo, I. M.; Chan, C. M., Simulation-based evolutionary optimization for energy-efficient layout plan design of high-rise residential buildings. *Journal of cleaner production* **2019,** 231, 1375-1388.

[18] Chen, X.; Huang, J.; Yang, H.; Peng, J., Approaching low-energy high-rise building by integrating passive architectural design with photovoltaic application. *Journal of Cleaner Production* **2019,** 220, 313-330.

[19] Chen, X.; Yang, H.; Peng, J., Energy optimization of high-rise commercial buildings integrated with photovoltaic facades in urban context. *Energy* **2019,** 172, 1-17.

[20] Giouri, E. D.; Tenpierik, M.; Turrin, M., Zero energy potential of a high-rise office building in a Mediterranean climate: Using multi-objective optimization to understand the impact of design decisions towards zero-energy high-rise buildings. *Energy and Buildings* **2020,** 209, 109666.

[21] Liu, J.; Wang, M.; Peng, J.; Chen, X.; Cao, S.; Yang, H., Techno-economic design optimization of hybrid renewable energy applications for high-rise residential buildings. *Energy Conversion and Management* **2020,** 213, 112868.

[22] Ekici, B.; Cubukcuoglu, C.; Turrin, M.; Sariyildiz, I. S., Performative computational architecture using swarm and evolutionary optimisation: A review. *Building and environment* **2019,** 147, 356-371.

[23] Cubukcuoglu, C.; Ekici, B.; Tasgetiren, M. F.; Sariyildiz, S., OPTIMUS: self-adaptive differential evolution with ensemble of mutation strategies for grasshopper algorithmic modeling. *Algorithms* **2019,** 12, (7), 141.

[24] Ekici, B.; Kazanasmaz, T.; Turrin, M.; Tasgetiren, M. F.; Sariyildiz, I. S. In *A Methodology for daylight optimisation of high-rise buildings in the dense urban district using overhang length and glazing type variables with surrogate modelling*, Journal of Physics: Conference Series, IOP Publishing: **2019**; p 012133.

[25] Ekici, B.; Kazanasmaz, Z. T.; Turrin, M.; Taşgetiren, M. F.; Sariyildiz, I. S., Multi-zone optimisation of high-rise buildings using artificial intelligence for sustainable metropolises. Part 1: Background, methodology, setup, and machine learning results. *Solar Energy* **2021,** 224, 373-389.

[26] Ekici, B.; Kazanasmaz, Z. T.; Turrin, M.; Taşgetiren, M. F.; Sariyildiz, I. S., Multi-zone optimisation of high-rise buildings using artificial intelligence for sustainable metropolises. Part 2: Optimisation problems, algorithms, results, and method validation. *Solar Energy* **2021,** 224, 309-326.

[27] Ekici, B.; Turkcan, O. F. S. F.; Turrin, M.; Sariyildiz, I. S.; Tasgetiren, M. F., Optimising High-Rise Buildings for Self-Sufficiency in Energy Consumption and Food Production Using Artificial Intelligence: Case of Europoint Complex in Rotterdam. *Energies* **2022,** 15, (2), 660.

INTRODUCTION — METHOD(S) | CHALLENGE(S)

LITERATURE REVIEW

	METHOD(S)	CHALLENGE(S)	
J1 RQ1	Optimising form-finding parameters in performative computational architecture	■ 17 Form-finding parameters ● 13 Performance aspects ▲ 12 Evolutionary algorithms 3 Swarm algorithms	

TOOL DEVELOPMENT AND PILOT STUDY

	METHOD(S)	CHALLENGE(S)	
J2 RQ2	PART A Developing Optimus tool using self-adaptive ensemble evolutionary algorithm	■ 30 Parameters ▲ 4 Optimisation algorithms	TEST 1
		■ 70 Parameters ▲ 4 Optimisation algorithms	TEST 2
J3 RQ3	PART B Preliminary results of multi-zone approach using pilot high-rise model	■ 100 Parameters ● 2 Daylight metrics ▲ 5 ANN models 1 Optimisation algorithm	TEST 3

METHODOLOGICAL FRAMEWORK

	METHOD(S)	CHALLENGE(S)	
J4 RQ4	PART A Introducing multi-zone optimisation (MUZO) methodology and prediction results of quad-grid and diagrid high-rise scenarios	■ 260 Parameters ● 2 Daylight metrics ▲ 20 ANN models	TEST 4
		■ 220 Parameters ● 2 Daylight metrics ▲ 20 ANN models	TEST 5
J5 RQ5	PART B Optimising high-rise scenarios using the predictive models with Optimus and validation of the MUZO methodology	■ 260 Parameters ● 20 Predictive models ▲ 3 Optimisation algorithms	TEST 4
		■ 220 Parameters ● 20 Predictive models ▲ 3 Optimisation algorithms	TEST 5

CASE STUDY

	METHOD(S)	CHALLENGE(S)
J6 RQ6	Optimising Europoint complex for self-sufficiency in energy consumption and food production using MUZO and Optimus	■ 117 Parameters ● 1 Self-sufficiency in energy 1 Self-sufficiency in food 1 Daylight metric ▲ 45 ANN models 13 Optimisation algorithms

CONCLUSIONS

J Journal
RQ Research question

■ Parameter ● Performance ▲ AI Method

TOC

2 Literature review

Optimising form-finding parameters in performative computational architecture

Chapter 2 has been published as: Ekici, B.; Cubukcuoglu, C.; Turrin, M.; Sariyildiz, I. S., Performative computational architecture using swarm and evolutionary optimisation: A review. Building and environment 2019, 147, 356-371. For consistency of the dissertation, the layout is adapted to fit the template, some typos are adjusted, and phrases are reworded without changing the content.

https://doi.org/10.1016/j.buildenv.2018.10.023

This chapter presents the literature review on optimising form-finding parameters in performative computational architecture that entails form generation, performance evaluation, and optimisation. A systematic review is conducted based on multiple databases to elaborate the trends for investigating well-performing design alternatives using optimisation algorithms in the architectural design domain. Therefore, the review focuses on studies involving form-finding parameters. One hundred studies are systematically reviewed, focusing on swarm and evolutionary optimisation algorithms frequently used in architectural design. The chapter concludes by presenting the gaps and needs considered while developing the optimisation tool and computational framework, focusing on form-finding parameters, performance evaluation, and optimisation applications. The outputs of this chapter are the inputs of the developed optimisation tool, form-finding parameters to be considered in parametric modelling in other tests and case study, and the utilisation of optimisation algorithms with comparison and replication.

Performative Computational Architecture using Swarm and Evolutionary Optimisation

A Review

ABSTRACT This study presents a systematic review and summary of performative computational architecture using swarm and evolutionary optimisation. The taxonomy for one hundred types of studies is presented herein that includes different sub-categories of performative computational architecture, such as sustainability, cost, functionality, and structure. Specifically, energy, daylight, solar radiation, environmental impact, thermal comfort, life-cycle cost, initial and global costs, energy use cost, space allocation, logistics, structural assessment, and holistic design approaches, are investigated by considering their corresponding performance aspects. The main findings, including optimisation and all the types of parameters, are presented by focussing on different aspects of buildings. In addition, usage of form-finding parameters of all reviewed studies and the distributions for each performance objectives are also presented. Moreover, usage of swarm and evolutionary optimisation algorithms in reviewed studies is summarised. Trends in publications, published years, problem scales, and building functions, are examined. Finally, future prospects are highlighted by focussing on different aspects of performative computational architecture in accordance to the evidence collected based on the review process.

KEYWORDS Performance-based design, building design, architectural design, computational optimisation, swarm intelligence, evolutionary algorithm

2.1 Introduction

Architectural design is a complex task. One of the most important reasons for its complexity is that multiple objectives affect the overall performance of the designed object [1]. In many cases, these objectives conflict with each other. In addition, each design is a unique task based on the problem, objectives, building program, constraints, client expectations, and the surrounding impacts owing to the built environment. For this reason, there are many "design-related parameters" to cope with the design process. Moreover, architectural design is a critical mission. Architects are responsible for creating living environments not only for human beings but also for all living creatures. Therefore, in the design process, the decisions require increased awareness for the anticipated consequences.

Conversely, design is an iterative process [2]. During this phase, the architect employs many design methods, such as sketching and physical as well as digital modelling, in order to feature the invention and revision cycles simultaneously. During this process, many criteria are considered, which correspond to the many requirements the final design is expected to satisfy. Such criteria regard several fields, from structural safety to climatic comfort, from energy efficiency to real estate values, etc. The ultimate goal is the identification of a design solution that satisfies at best many different (and sometime conflicting) objectives. Most of these objectives are highly affected by the decisions taken during the early design phase.

Performance-based design (PBD) has become a vital approach to satisfy many objectives. Kolarevic [3] underlined the importance of PBD as a guiding design principle. Among several possible approaches, this study focuses on a specific framework, which was presented by Sariyildiz [4] in order to support the design process. The presented framework is called performative computational architecture (PCA). This framework consists of three main phases as illustrated in Fig. 2.1. These are form generation, performance evaluation, and optimisation. Therefore, the main purpose of PCA is to investigate the most desirable geometry that satisfies performance-related goals in the conceptual design stage. In this study, journal articles associated with PCA are reviewed. Section 2.1.1 highlights the focus based on the relevant search of journal articles in the literature. Section 2.1.2 presents review articles that have similar focus, and underlines the differences with this study.

TOC

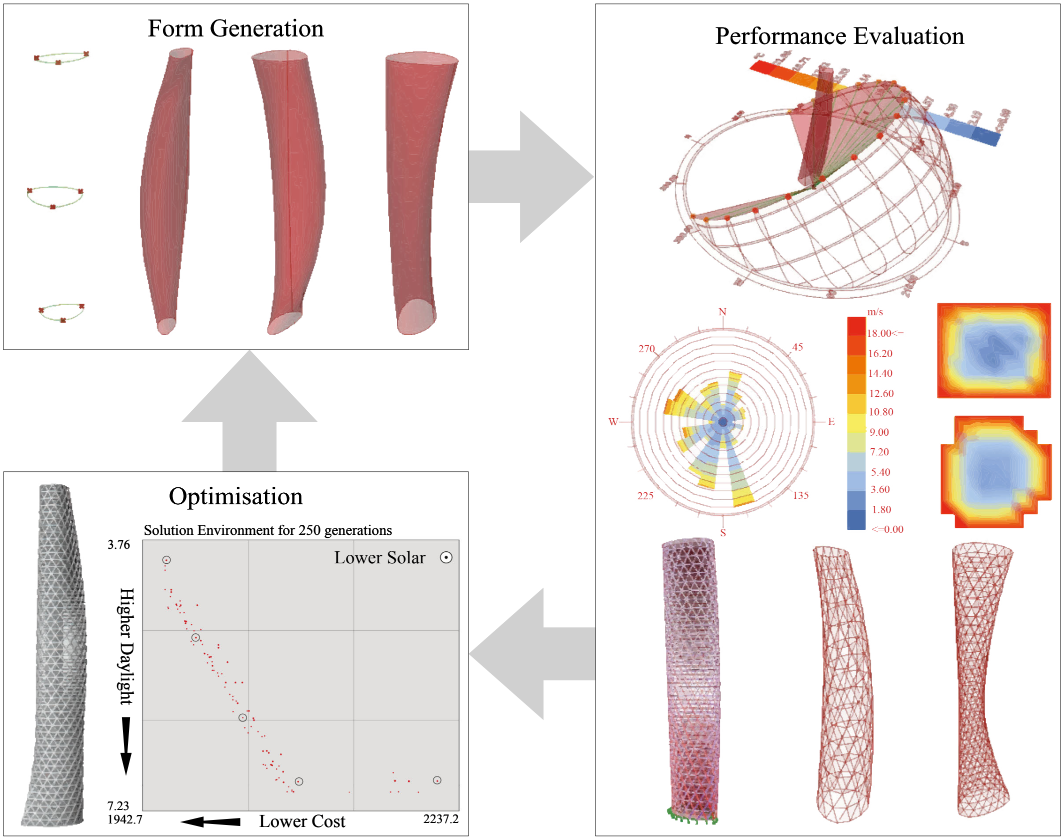


FIG. 2.1 PCA framework

2.1.1 Focus of this review

This review focuses on studies that match the PCA framework. Dealing with different geometric configurations as design alternatives (obtained from computational processes for form generation) is crucial to this focus. Optimising the designs by means of geometric variations in the early design stages is essential in architectural design. This clearly differentiates the use of optimisation in architectural design from the use of optimisation in engineering. In this sense, the architectural shape is mostly used as a set of specific boundaries within which a search for good engineering solutions is conducted. The shape is not typically modified as is done during the architectural explorations in PCA. As such, this review considers only the studies in which architectural geometric variations are (also) included (and it is not concerned with purely engineering optimisation).

Moreover, this review is concerned only with specific optimisation methods. It focuses on swarm and evolutionary computation (SEC). There are two reasons for this choice. First of all, direct search methods require expensive computational time to handle many parameters in the optimisation problem [5,6]. Secondly, metaheuristics can suggest near-optimal solutions with many design parameters within a reasonable time [7]. Swarm intelligence (SI) and evolutionary computation (EC) are two powerful optimisation methods in metaheuristics. SI uses intelligent multi-agent systems inspired by the behaviour of social swarms [8]. Conversely, EC uses procedures inspired by the biological evolution of the Darwinian theory [9].

Numerous publications were analysed within a broad spectrum of thematic areas by considering the PCA framework and SEC. To identify relevant studies, keywords such as “building design”, “architectural design”, “evolutionary algorithm”, “evolutionary computation”, “swarm intelligence”, and “swarm optimisation” were used. During this search, Science Direct, Scopus, and Thomson Reuters, were employed as databases. To investigate the field in-depth, there was no time limitation. The final cut-off date for published studies was 26 August 2018. This broad search led to the collection of a relevant number of publications. From this collection, a subset of journal articles was selected according to the defined criteria. These studies

- Include all three phases of PCA (form generation, performance evaluation, and optimisation)
- Explicitly deal with architectural form-finding (on real and/or hypothetical architectural designs)
- Include optimisation processes based on swarm and/or evolutionary computation
- Must be published as journal (not conference) articles (because most conference studies lack fundamental information)
- Could consider any performance criteria (there was no selection based on specific performance criteria)

An initial analysis of the selected sub-set, led to the identification of additional criteria that were followed to review the selected papers, as follows,

- Most building design problems could be analysed by categorising them in accordance to the layout, skin, and overall building shape
- Most holistic approaches obviously integrate several design decision steps that have to be analysed by taking this integration into account

2.1.2 Focus of previous review articles

Several reviews have been published by other authors during the past few decades. Some of them share a similar approach with this review, but in their searches/evaluations they have not included the recent decades and/or have focussed only on some of the three phases of PCA. Some of them focussed on the categorisation and performance of optimisation algorithms. Most of them focus only on specific building performance criteria. All of them include relevant information and different perspectives.

In an early study in 1980, Radford and Gero [10] discussed simulation, generation, and optimisation methods for supporting architectural design decisions. Touloupaki and Theodosiou [11] presented a recent review on the combined use of parametric modelling, performance simulations, and optimisation algorithms. Based on several examples, their review provides valuable highlights on the potentials and limitations of the current state-of-art. However, it does not include a systematic analysis of trends, nor of used design variables and objectives. Negendahl [12] focussed on building performance simulations. In turn, Machairas, Tsangrassoulis and Axarli [6], reviewed optimisation algorithms for building design by considering tools, objectives, and performance assessments. In another study, Nguyen, et al. [13], overviewed simulation-based optimisation methods for building performance analyses based on the discussion of the major challenges. Concerning the building envelope, Huang and Niu [5] reviewed numerous studies to compare popular optimisation algorithms. When the focus is on specific performance domains, subjects such as the efficient spatial planning, energy efficiency, daylight, etc. may constitute relevant examples. Concerning the layout configuration, Dutta and Sarthak [14] compared applications of EC for architectural space planning. For sustainable building design, Evins [15] reviewed the application of computational optimisation by considering different research branches. In another study, Attia, et al. [16], investigated potential challenges and opportunities for the integration of optimisation tools in net-zero energy buildings (NZEBs). Shi, et al. [17], reviewed simulation-based design generation and optimisation in order to discuss their applications on energy-driven urban design at the district scale. Cui, et al. [18], reviewed multi-objective optimisation applications for environmental protection fields (such as optimisation for energy saving and for emission, and cost reductions).

Based on users surveys and literature reviews, Tian, et al. [19], focussed on the application of building energy simulations and optimisations for passive building designs. Kheiri [20] highlighted the potentials of different optimisation methods to shape energy-efficient architectural building geometries and envelopes. Shi, et al. [21], focussed on energy performance by analysing several optimisation methods, including the types of algorithms, the design objectives and variables, and the energy simulation engines. Eltaweel and Yuehong [22] focussed on the parametric design for daylight and solar radiation. In contrast to prior reviews, the review presented herein focuses on form generation and on performance evaluation and optimisation by offering a systematic analysis and categorisation of design variables and design objectives, without being confined to specific performance criteria.

2.2 Performative computational architecture and review taxonomy

This section presents the three phases of PCA in depth. As previously mentioned, the first phase is form-finding, which corresponds to the form generation in this iterative process. The second phase is performance evaluation, which focuses on objectives that are desired to be satisfied in form-finding. The final phase is optimisation, which uses search method to identify satisfactory design alternatives in a systematic way. The three phases are iteratively looped. Section 2.2.1 introduces the form-finding phase of PCA. Section 2.2.2 explains the role of performance evaluation in this framework. Section 2.2.3 focuses on SI and EC as part of the optimisation phase of PCA. Finally, Section 2.2.4 introduces PCA taxonomy.

2.2.1 Form-finding

Early examples of form-finding studies have focussed on structure, especially for shell designs. Antoni Gaudi is accepted as one of the pioneer architects of this field, based on his work on hanging-chain models [23]. Therefore, form-finding is defined as a forward process controlled by parameters to discover an optimal geometry of a structure that is in static equilibrium subject to a specific design loading scheme [24]. From the standpoint of structural form-finding, several definitions can be found in addition to those described in [25,26].

In this study, the notion of form-finding is beyond the structural performances alone, and is defined as the architectural design exploration aiming to satisfy predetermined building performance aspects via computational optimisation in order to provide sufficient information to the decision-makers. This includes performance aspects other than structural performance. For the sake of clarity, the shape of the building affects many performances, such as energy consumption, daylight usage, layout configuration, functional accessibility, shading performance, solar gain, acoustics, and others. In this context, form-finding corresponds to one of the most crucial steps in the conceptual design process. The reason is attributed to the fact that this step comprises the decisions on the determination of the mass and shape of the overall form of the design. Therefore, form-finding outputs are inputs for all subsequent steps in the design process, in the subsequent construction phase, and throughout the building's life-cycle.

2.2.2 Performance evaluation

With the recent developments in digital technology, the predictions and numeric assessments of performance aspects can be integrated into the architectural design process in order to investigate how well the design eventually meets the requirements. This regards all the design phases, and it is especially important in the conceptual stage. Despite the importance of the decisions taken in the early phase, current practice lacks numeric assessments in the conceptual design phase.

Broadbent [27] pointed out that the amount of a priori knowledge available at the beginning of each design process highly depends on the design case, and is quite limited when innovation is involved in the process. Hubka and Eder [28] emphasised that the design has traditionally been conducted using intuition, know-how, and judgment. This highlights the need for measuring and numerically assessing the capacity of the design in satisfying the various requirements and supporting the exploration of design alternatives by means of multidisciplinary measurable performance values as guiding criteria.

Turrin [29] emphasised that geometry has an enormous impact on the realisation of performance-related goals. Owing to the number of parameters, many design alternatives exist in the search space [30]. For this reason, discovering feasible and desirable design solutions is a complicated task during the performance evaluation phase. To support this process, computational optimisation techniques have proven to be relevant. In fact, owing to the size of the solution space, a systematic performance assessment by the designer for each desirable design solution is generally impossible owing to time and other restrictions. Furthermore, a systematic exploration of the solution space that aimed at selecting a subset of solutions is challenging when is simply left to the intuition of the designer.

2.2.3 Swarm and evolutionary computation for optimisation

In the domain of architectural design, metaheuristics constitute one of the most extensively used optimisation methods, and corresponds to the third phase of PCA [6,15,31]. These search algorithms are capable of dealing with continuous and discrete parameters in large parameter spaces, and they also avoid local minima and maxima. Moreover, when compared to other direct search methods, metaheuristics are capable in presenting near-optimal results in a reasonable time [7].

SI and EC are based on different search strategies inspired by nature. In the EC procedure, individuals with decision variables in D dimensions are encoded into chromosomes to obtain an initial population. At each generation, pair(s) of individuals from the population are chosen and mated. These individuals are then crossed over to generate new solutions referred to as offspring or children. Some individuals are mutated to escape from local minima and maxima. Ultimately, the offspring population is combined with the parent population to select new individuals for the next generation. The genetic algorithm (GA) proposed by Holland and Goldberg [32], and the differential evolution (DE) presented by Storn and Price [33], can be used in EC.

On the other hand, SI focuses on the interactions of individuals with each other and their environment. For this reason, SI uses societies, such as ants, wasps, termites, bees, schools of fish, flocks of birds, and herds of land animals. An SI algorithm typically consists of many individuals. Simple behavioural rules direct the interactions among the individuals in D dimensions. As a result of the overall behaviour of the swarm system, there are consequences to the self-organising group behaviour. Particle swarm optimisation (PSO) founded by Eberhart and Kennedy [34], ant colony optimisation (ACO) suggested by Dorigo, et al. [35], can also be used in SI. As an example, procedures of generic EC and SI are illustrated in Fig. 2.2.

2.2.4 Review taxonomy

One hundred journal articles relevant to the focus of this review were identified. In order to investigate these papers systematically, a PCA taxonomy was defined as shown in Fig. 2.3. The main categories of this taxonomy were sustainability, cost, functionality, and structure. In addition, several sub-categories were also determined according to performance objectives as shown in Fig. 2.3. In the following chapters, each sub-category is explained in detail.

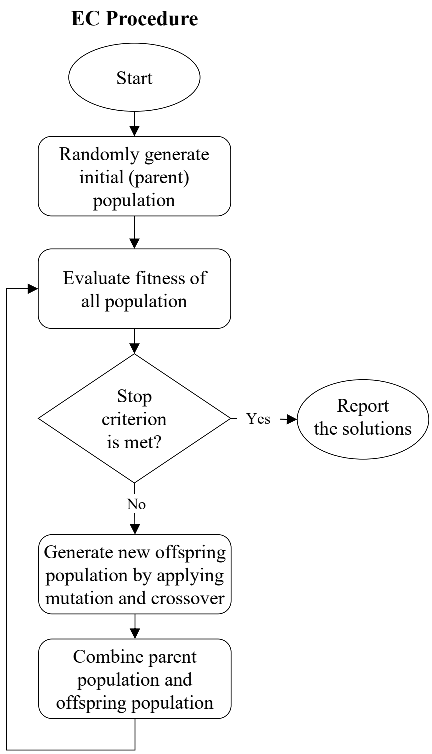


FIG. 2.2 EC and SI procedures

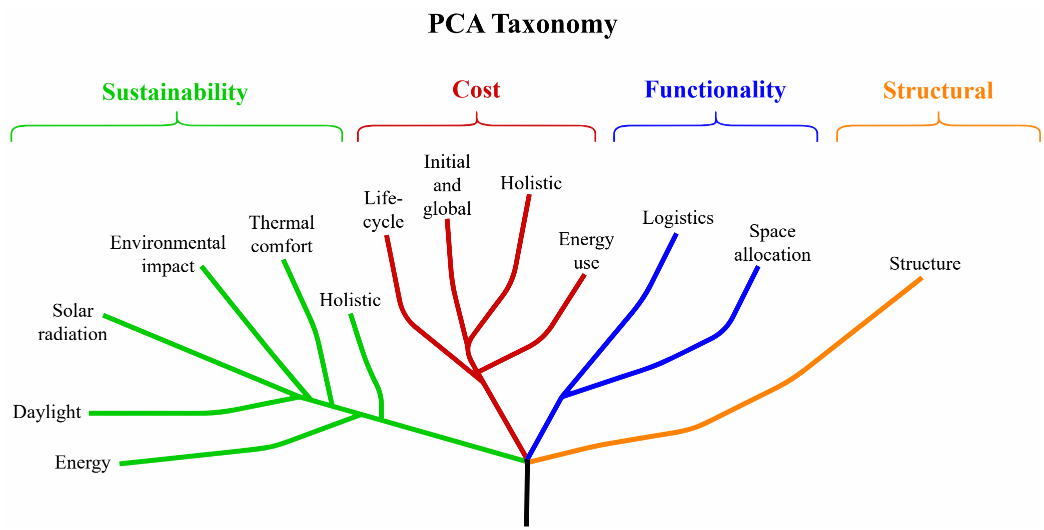


FIG. 2.3 Taxonomy

TOC

2.3 Sustainability

2.3.1 Energy

As an early example that focussed on the skin of the building, Caldas and Norford [36] minimised the annual energy consumption for office buildings using GA-based design tools. Afterwards, Caldas, et al. [37] minimised the annual energy consumption with GA for a school building. Wetter and Wright [38] used the discrete armijo gradient (DAG), GA, coordinate search (CS), Hooke-Jeeves (HJ), Nelder-Mead (NM), PSO with HJ, and other variants of PSO, to minimise the annual energy consumption for an office building. Lee [39] combined GA and computational fluid dynamics (CFD) to minimise energy supplied by HVAC for heating and cooling in office buildings. More recently, Bucking, et al.[40], minimised the net annual energy consumption for a net-zero energy house design using modified EA and PSO. Ramallo-González and Coley [41] minimised the heating and cooling demands of residential building a using covariance matrix adaptation (CMA), evolution strategy (ES), sequential assessment (SA), and the canonical form of the GA.

As one of the early examples of multi-objective optimisation, Naboni, et al. [42], used the non-dominated sorting genetic algorithm II (NSGA-II) to minimise heating, cooling, and lighting demands for a residential building. Méndez Echenagucia, et al. [43], also used NSGA-II to minimise the energy need for heating, cooling, and lighting in office buildings. Xu, et al. [44] examined the trade-offs between cooling and heating loads using NSGA-II for envelope design of an office building. Wright and Alajmi [45] minimised the building's energy consumption using the GA for the building envelope design of an office building. Delgarm, et al. [46], minimised the annual cooling, heating, and lighting electricity consumptions using single and multi-objective approaches using PSO. In turn, Delgarm, et al. [47], minimised the annual cooling and lighting electricity consumptions considering two different optimisations using single-objective GA and NSGA-II. Si, et al. [48] minimised the annual energy consumption using the HJ, multi-objective genetic algorithm II (MOGA-II), and the multi-objective PSO (MOPSO) for an office building envelope. Li, et al. [49], used MOPSO and artificial neural networks (ANN) to minimise the energy consumption for residential buildings. Bre and Fachinotti [50] used NSGA-II to examine trade-offs between heating and cooling demands for residential buildings, as well.

Bamdad, et al. [51], minimised the annual energy consumption of commercial buildings using several optimisation algorithms, such as ACO, NM, and hybrid PSO variants. Chen and Yang [52] minimised the heating, cooling, and lighting energy demands of high-rise residential buildings using NSGA-II for formulating the bi-objective and three objective optimisation problems. Recently, Bamdad, et al. [53], minimised the energy use by considering low, base, and high simulation scenarios for office buildings using ant colony optimisation algorithm for mixed variables (ACOMV), and proposed a modified ACOMV.

By focusing on building shapes, Caldas [54] optimised the energy consumption, energy use intensity, thermal, and daylight performances, and the initial cost of materials using GA and Pareto GA in office and school buildings. Lin and Gerber [55] presented evolutionary energy performance feedback for a design (EEPFD) approach using multi-disciplinary design optimisation (MDDO). Related to this work, Lin and Gerber [56] minimised the energy use and maximised the spatial programming compliance score with the net present value for several building cases using MOGA. Another recent work of Gerber and Lin [57] included additional qualitative data-driven by human designers, and discussed the importance of EEPFD in the conceptual phase. Recently, Li, et al. [58], minimised the total energy consumption by focusing on the heating, cooling, and lighting demands of a school building that employed GA. Moreover, Bizjak, et al. [59], first minimised the heating and cooling loads, and then maximised the heat gain using single-objective DE for residential building.

Apart from these, several other studies also included the energy aspects, but these studies are discussed in other sections. Futrell, et al. [60], Negendahl and Nielsen [61], Chen, et al. [62], and Chatzikonstantinou and Sariyildiz [63] evaluated and explained the effects of daylight. Yi [64] presented the impacts of solar radiation. Azari, et al. [65], stated the environmental impact. Magnier and Haghighat [66], Kasinalis, et al. [67], Yu, et al. [68], Zhang, et al. [69], Lin, et al. [70], and Gou, et al. [71], investigated thermal comfort. Dhariwal and Banerjee [72], and Harkouss, et al. [73], analysed the life-cycle cost. Znouda, et al. [74], Talbourdet, et al. [75], Wright, et al. [76], Brownlee and Wright [77], Yang, et al. [78], Rafiq and Rustell [79], and Chang and Shih [80], presented evaluations on the initial and global costs. Michalek, et al. [81], and Baušys and Pankrašovaité [82] studied energy use cost. Finally, Menges [83] and Yang, et al. [84] discussed structure.

2.3.2 Daylight

In an early study that focussed on the skin of the building during daylight, Turrin, et al. [85] maximised the daylight factor and minimised the solar incidence and structural weight of a parametric long-span roof using GA. Rakha and Nassar [86] minimised the daylight uniformity ratio for gallery building using GA. Gagne and Andersen [87] also utilised GA for maximising non-conflicting illuminance goals. Conversely, authors also applied multi-objective micro-GA for maximising illuminance and minimising glare objectives. Futrell, Ozelkan and Brentrup [60], employed PSO using the construction coefficient and the HJ algorithm, while they maximised the pi scores from hourly illuminance outcomes, and minimised the thermal performance based on the sum of annual hourly energy consumption for envelope design.

Negendahl and Nielsen [61] optimised daylight performance, capital cost, building energy use, and thermal requirements for a folding façade design using the strength Pareto evolutionary algorithm 2 (SPEA-2). Futrell, et al. [88], utilised several algorithms, such as NM, HJ, and variants of PSO to maximise the daylight performance for a classroom. Chen, et al. [89], used NSGA-II to minimise the daylight and thermal discomfort times. Recently, Chatzikonstantinou and Sariyildiz [63] minimised the trade-off between the energy consumption and artificial light dependence for an office building using NSGA-II. In order to characterise alternatives with good performance, authors presented an auto-associative machine learning framework.

From the viewpoint of building shape, Chen, Janssen and Schlueter [62], maximised daylight and minimised the cooling energy consumption using NSGA-II of a parametric building. In addition to these, several studies considered daylight as a performance aspect as well. Caldas [54] presented details on energy considerations. Zhang, Bokel, van den Dobbelsteen, Sun, Huang and Zhang [69], explained aspects of thermal comfort. Chang and Shih [80] discussed initial and global costs. Su and Yan [90] stated and evaluated logistics. Finally, Yang, Ren, Turrin, Sariyildiz and Sun [84], considered and analysed the structure.

2.3.3 Solar radiation

Based on the skin of the building, Bizjak, et al. [91], used a self-adaptive differential evolution (DE) algorithm to maximise solar irradiation. For the sake of the shape of the building, Liu, et al. [92], performed PSO to minimise the solar gain and maximise the area of the residential buildings in an urban setting. Oliveira Panão, et al. [93], also used GA for maximising the absorption of solar radiation in the winter season and minimise it during the summer in urban forms. Kämpf and Robinson [94] maximised the solar energy potential using two different algorithms, namely CMA-ES, and the hybrid differential evolution (HDE), by focussing on three different cases. In comparison, Kämpf, et al. [95], minimised irradiation offset by thermal losses, while they maximised building volumes using MOEA. Yi [64] used MOEA for solar radiation and energy consumption in high-rise office buildings. Zhang, et al. [96], used GA to maximise the total radiation as a function of the shape efficiency, and to minimise the shape coefficient for a community centre. More recently, Vermeulen, et al. [97], maximised solar radiation in an urban context using EA for winter, equinox, and summer times.

From the viewpoint of the layout, António, et al. [98], also maximised energy received per building in an urban context by considering different building amounts using GA. Yi and Kim [99] minimised the solar radiation of a set of residential blocks using GA. In another study, Vermeulen, et al. [100], maximised the solar energy received by each building using an evolutionary algorithm (EA) for high-rise buildings.

In addition to these, several studies were associated with solar radiation but are explained in other sections. Turrin, Von Buelow and Stouffs [85], considered daylight. Menges [83] explained the environmental impact, whereas other cases considered structure.

2.3.4 Environmental impact

Wang, et al. [101], focussed on the building’s skin to minimise life-cycle costs and life-cycle environmental impacts using MOGA. Rapone and Saro [102] minimised carbon emissions for a single office zone using PSO considering different cities. More recently, Azari, Garshasbi, Amini, Rashed-Ali and Mohammadi [65], utilised NSGA-II for minimising the environmental life-cycle impact and operational energy use for set objectives. Additionally, authors also utilised GA for a single objective environmental life-cycle impact optimisation problem with ANN.

With regard to the shape of the building, Wang, et al. [103], used MOGA to examine the trade-offs between the life-cycle cost and life-cycle environmental impact for a green building design. Following this, Wang, et al. [104], addressed a similar problem using MOGA to minimise the life-cycle environmental impact and life-cycle cost. Menges [83] optimised environmental criteria, such as block ventilation, covered outside space, outer solar radiation, unit ventilation, solar radiation per unit, circulation, and unit count. More recently, Huang, et al. [105], minimised the building shadow area in an urban setting as an environmental impact form, and maximised the building floor area to reach a satisfactory building mass using GA. McKinstray, et al. [106], minimised the carbon impact and embodied carbon of a frame portal building using MOEA.

There are other publications that considered environmental impact objectives, as well. Li, Pan, Xue, Jiang and Mao [49], referred to in Section 2.3.5 considered on thermal comfort. Karatas and El-Rayes [107], Liu, et al. [108], and Hester, et al. [109], discussed life-cycle cost, as outlined in Section 2.4.1.

2.3.5 Thermal comfort

All the reviewed publications in this section focussed on the building's skin. As an early example, Magnier and Haghighat [66] used simulation-based ANN with NSGA-II to minimise the average absolute thermal comfort and annual energy consumption. Kasinalis, Loonen, Cóstola and Hensen [67], also used NSGA-II to examine the trade-offs between the thermal discomfort and annual primary energy consumption for seasonally adaptable façade designs in office buildings. Yu, Li, Jia, Zhang and Wang [68], optimised the annual energy consumption and the percentage of thermal discomfort hours with the use of NSGA-II in residential cases. Furthermore, Li, Pan, Xue, Jiang and Mao [49], used NSGA-II, MOPSO, MOGA, and multi-objective DE (MODE), to optimise the total percentage of cumulative time with discomfort, life-cycle cost and carbon dioxide equivalence for residential cases.

Zhang, Bokel, van den Dobbelsteen, Sun, Huang and Zhang [69], optimised a school model using SPEA-2 in order to minimise the energy use and summer discomfort time, while maximising useful daylight illuminance (UDI). Lin, Zhou, Yang and Li [70], also maximised thermal comfort and minimised energy consumption using MOGA with multi-linear regression (MLR) and ANN. Sghiouri, et al. [110], minimised discomfort hours using NSGA-II in residential buildings. Gou, Nik, Scartezzini, Zhao and Li [71] maximised the annual indoor thermal comfort and minimised building energy demand using NSGA-II and ANN for residential buildings. Chen, Yang and Sun [89], also considered the thermal comfort performance objective, but their findings are presented in the daylight section.

2.3.6 Holistic sustainability

Finally, holistic approaches are discussed which considered several steps to reach a sustainable design. In an early study on the building's skin, Ercan and Elias-Ozkan [111] focussed on the atrium design. In the first stage, authors minimised solar irradiation of the building using GA. In the second stage, they focussed on the façade shading device to minimise the standard deviation of the daylight factor and annual solar irradiation. Recently, Ferrara, et al. [112], optimised a single-zone classroom model using PSO and focussed on different cities with various orientations. In the first step, authors minimised the total energy demand. Thereafter, they maximised thermal and visual comfort.

Based on the building shape, Youssef, et al. [113], optimised the energy consumption of office buildings using GA in two steps. First, authors optimised the shape of the building using shape grammar design rules. Secondly, they optimised the façade design for the integration of photovoltaic panels.

For the sake of the building layout, Sleiman, et al. [114], firstly focussed on the layout configuration aspects using EA. Subsequently, authors considered two major performance objectives, namely, the energy performance and life-cycle cost for a healthcare facility. Similarly, Dino and Üçoluk [115] synthesised the building's space layout performance with energy and daylight aspects. In the first step, authors optimised the unique fitness function by considering several layout aspects using EA. Thereafter, they used NSGA-II for daylight autonomy and total energy.

2.4 Cost

2.4.1 Life-cycle cost

By focussing on the building's skin, Tuhus-Dubrow and Krarti [116] minimised the life-cycle cost using GA, PSO, and sequential search (SS) algorithms for five different climates. Conversely, Bichiou and Krarti [117] minimised life-cycle cost using GA, PSO, and SS. Using a different approach, Gengembre, et al. [118], used PSO with Kriging metamodeling for the minimisation of the life-cycle cost. In comparison to

TOC

previous studies, Karatas and El-Rayes [107] considered the minimisation of both the life-cycle cost and operational environmental cost, and the maximisation of social quality of life using MOGA for a single-family house. Afterwards, Karatas and El-Rayes [119] used MOGA to minimise the life-cycle cost and maximise social quality. In another multi-objective optimisation approach, Liu, Meng and Tam [108], minimised the life-cycle cost and life-cycle carbon emissions using MOPSO. Ferrara, et al. [120], minimised the global cost over the life-cycle using PSO for a single-family house.

Dhariwal and Banerjee [72] proposed an approach using fractional factorial design and response surface methods to optimise the life-cycle cost. To validate the proposed approach, authors minimised the incremental life-cycle cost using GA and minimised the both life-cycle cost and energy use intensity using NSGA-II. More recently, Hester, Gregory, Ulm and Kirchain [109], minimised the life-cycle cost and life-cycle impact in order to explore the building's design space by comparing GA results, sequential specifications, and unguided specification algorithms. Harkouss, Fardoun and Biwole [73] minimised an auxiliary electric heater with a pump, thermal demands for cooling and heating, exports, and life-cycle costs using NSGA-II for NZEBs.

In addition to these studies, several publications included the life-cycle cost as well. However, these studies are mentioned in other sections. Lin and Gerber [55], Lin and Gerber [56], and Gerber and Lin [57], explained and discussed energy considerations. Wang, Zmeureanu and Rivard [101], Wang, Rivard and Zmeureanu [103], and Wang, Rivard and Zmeureanu [104], are referred to in the environmental impact section. Finally, Li, Pan, Xue, Jiang and Mao [49], are cited in the thermal comfort section.

2.4.2 Initial and global costs

In consideration of the skin of the building, Znouda, Ghrab-Morcos and Hadj-Alouane [74] used GA to minimise the global monetary cost for four different economic scenarios based on gas and electricity. For the investigation of trade-offs between the minimisation of the construction cost and energy need objectives, Talbourdet, Michel, Andrieux, Millet, Mankibi and Vinot [75], used NSGA-II for an office building. Wright, Brownlee, Mourshed and Wang [76], also optimised a commercial building in order to minimise the energy use and capital cost objectives using NSGA-II by considering several optimisation experiments. For the same trade-off, Brownlee and Wright [77] also used NSGA-II but with surrogate models based on radial basis functions (RBFs). More recently, Yang, Lin, Lin and Tsai [78], used NSGA-II in three different analyses approaches. The first approach minimised both the envelope

TOC

construction cost and energy performance. The second approach minimised the objectives that were considered in the first analysis and maximised the window opening rate. The third analysis focussed on the same objective functions used in the second analysis scheme but for different climatic zones in Taiwan.

For the sake of the building shape, Chang and Shih [80] integrated dynamic programming and GA to minimise the construction cost with the energy cost, and maximised the area of visual view with daylight illumination, for a residential building. Rafiq and Rustell [79] minimised the structural cost, energy loss, and area loss objectives, using interactive visualisation clustering GA in the case of a commercial building.

In an early study that focussed on building's layout, Gero and Kazakov [121] minimised the layout cost based on travel distances and space relations using GA for office and hospital building cases. Apart from these, Caldas [54] and Negendahl and Nielsen [61] considered the initial and global costs, respectively, but they are mentioned and referred to in the energy and daylight sections.

2.4.3 Energy use cost

In this part, there are only two studies that focussed on the building layout. Michalek, Choudhary and Papalambros [81], minimised the heating cost, cooling cost, lighting cost, wasted space, hall size, and access way size, using GA and simulated annealing (SA) for residential buildings. Baušys and Pankrašovaité [82] minimised the heating cost, lighting cost, wasted space, doorways, and hallways using improved GA in the case of a residential building.

2.4.4 Holistic cost

In consideration of the building skin, Evins [122] proposed a multi-level optimisation framework by dividing the design and operation of a building into three phases: building, plant, and operational levels. For the building and plant levels, the author used NSGA-II to minimise the annual carbon emissions and initial capital cost. At the operational level, a mixed-integer programming approach was used to minimise the annual running costs.

For the sake of the building shape, Khajehpour and Grierson [123] integrated EC and colour filtering in a high-rise office building. In particular, the first step of the study applied optimisation techniques to minimise the capital, and operating costs, and to maximise the income revenue using a multi-criteria genetic algorithm (MCGA). The second step focussed on the determination of profit and safety potentials using colour filtering.

2.5 Functionality

2.5.1 Space allocation

As an early example for a building layout, Rodrigues, et al. [124], presented an optimisation framework. Authors considered adjacency, space overlap, opening overlap, and orientation, floor dimensions, compactness, and overflow using a hybrid evolutionary technique and ES with a stochastic hill climbing (SHC). To validate the proposed method, the authors applied the framework used to a residential layout problem [125]. As another example of residential layout, Song, et al. [126], used implicit redundant representation GA and simple GA for maximising symmetry, structural safety, stair connectivity, and façade exposure. More recently, Yazici [127] minimised the total built area in urban layouts using the EA and parametric design environment.

2.5.2 Logistics

As a manifestation of early work conducted in this area, Jo and Gero [128] optimised the interactions between interrelated spaces and the travel cost between the space elements with GA in an office layout case. More recently, Wong and Chan [129] optimised the adjacency preference matrix, adjacency limitations caused by physical and budget constraints, range of relative ratios between spaces, and the number of functions that can contribute acceptable designs using EA for a residential building. Focussing on health campuses, Güleç Özer and Şener [130] used GA to find an optimal route of users in functionally complex buildings.

TOC

Related to the healthcare facility, Su and Yan [90] optimised a nursing unit layout by maximising daylight illuminance and by minimising travel distances of nurses using GA. Dino [131] used EA to optimise unique fitness functions based on size, absolute dimension, compactness, jaggedness, convexity of space, as well as the façade, floor, neighbourhood, and separation criteria. The author focussed on a three-dimensional library building layout problem to implement the developed method. Cubukcuoglu, et al. [132], maximised accessibility, visibility, and wind protection objectives by proposing a multi-objective harmony search (MOHS) algorithm for an urban context. Authors also compared the results of MOHS with the self-adaptive differential evolution multi-objective (jDEMO) algorithm. More recently, Bahrehmand, et al. [133] optimised the overflow quality, topological quality, spatial quality, and user rating, with the use of an interactive EA in the case of a museum building. Gero and Kazakov [121] was also related to the logistics aspect but was mentioned in initial and global costs.

2.6 Structure

By focussing on the building's skin, Turrin, Von Buelow and Stouffs [85], used GA to minimise the weight of the dome design with an acceptable deformation for a semi-spherical structure. Authors combined the structural performance with the architectural form-finding process. In another study, Li [134] minimised discontinuous edges on the façade of a museum design using a GA-based split edge algorithm and an SS-based split edge algorithm. More recently, Yang, Ren, Turrin, Sariyildiz and Sun [84] optimised the roof of the sports building in order to maximise UDI, and minimise energy use and structural mass using NSGA-II.

For the sake of the building shape, Menges [83] used EA to optimise the morphologic criteria, such as the floor area, envelope heights, envelope slope, unobstructed view axes, incident solar radiation, and interior thermal loading. More recently, Elshaer, et al. [135], used ANN to minimise the mean drag coefficient for the first case and minimised the standard deviation of the lift coefficient for the second case using GA in high-rise buildings.

2.7 Review results

To highlight the correspondence between architectural geometry and performance, the form-finding parameters, and their corresponding performance objective(s) with respect to the building were presented in Table 2.1. To sum up, the distribution of each sub-category within the total number of performance objectives presented in the one hundred reviewed papers considered herein are shown in Fig. 2.4. All decision variables (both related and non-related with form-finding), and SEC methods used for each performance objective, are presented in Table 2.2. In relation to Table 2.1, the relationship between form-finding parameters used for each performance objectives in reviewed papers are listed in Fig. 2.5, and the total usage amount of form-finding parameters, such as window-to-wall ratio (WWR), and orientation, are shown in Fig. 2.6. The total usage of building topics are also listed in Fig. 2.7. All trade-offs between each performance objective used in bi-objective, three objective, and many-objective optimisation problems are illustrated in Fig. 2.8. Finally, other relevant information, which is not mentioned above, is summarised in Fig. 2.9. Since different methods are used in some papers, some graphs are presented as pie charts. This also includes the distribution of objectivity (where the term “many objectives” indicates a minimum of four objectives [136]).

TABLE 2.1 Overview of form-finding parameters, performance objectives and topic for each reviewed paper

			Form-finding parameters																
			Adjacency matrix	Number of buildings	Building's shape	Ceiling design	Façade design	Floor height	Light shelf	Orientation	Roof's shape	Roof's structure	Shading	Space dimensions	Space location	2D / 3D grid matrix	Window dimensions	Window location	Window-to-wall ratio
Ref.	Author(s)	Year	x1	x2	x3	x4	x5	x6	x7	x8	x9	x10	x11	x12	x13	x14	x15	x16	x17
[121]	Gero and Kazakov	1998													x				
[128]	Jo and Gero	1998													x				
[36]	Caldas and Norford	2002															x		
[81]	Michalek et al.	2002	x											x	x		x		
[37]	Caldas et al.	2003									x		x				x		
[123]	Khajehpour and Grierson	2003			x														x
[38]	Wetter and Wright	2004								x			x				x		
[82]	Baušys and Pankrašovaité	2005												x					
[101]	Wang et al.	2005								x									x
[103]	Wang et al.	2005			x					x			x						x
[104]	Wang et al.	2006			x								x						x
[39]	Lee	2007															x		
[92]	Liu et al.	2007		x	x			x							x				
[74]	Znouda et al.	2007					x						x						x
[54]	Caldas	2008					x	x			x		x	x			x		
[93]	Oliveira Panão et al.	2008		x	x					x									
[129]	Wong and Chan	2009	x																
[94]	Kampf and Robinson	2010			x					x						x			
[95]	Kampf et al.	2010			x					x									
[66]	Magnier and Haghighat	2010																	x
[116]	Tuhus-Dubrow and Krarti	2010			x								x						
[117]	Bichiou and Krarti	2011			x					x			x						x
[86]	Rakha and Nassar	2011				x													
[85]	Turrin et al.	2011									x	x	x						
[87]	Gagne and Andersen	2012					x						x					x	x
[118]	Gengembre et al.	2012											x						x
[134]	Li	2012										x							
[83]	Menges	2012					x					x							
[102]	Rapone and Saro	2012											x						x
[40]	Bucking et al.	2013					x			x	x		x						x
[130]	Güleç Özer and Şener	2013	x											x	x				
[107]	Karatas and El-Rayes	2013													x			x	x
[124]	Rodrigues et al.	2013	x											x	x		x		

	Performance objectives													Topic		
	Energy	Daylight	Solar radiation	Environmental impact	Thermal comfort	Holistic sustainability	Life-cycle cost	Initial and global costs	Energy use cost	Holistic cost	Space allocation	Logistics	Structure	Building skin	Building shape	Building layout
	y1	y2	y3	y4	y5	y6	y7	y8	y9	y10	y11	y12	y13	z1	z2	z3
								●				●				□
												●				□
	●													□		
	●								●							□
	●													□		
										●					□	
	●													□		
	●								●							□
				●			●							□		
				●			●								□	
				●			●								□	
	●													□		
			●												□	
	●							●						□		
	●	●						●							□	
			●												□	
												●				□
			●												□	
			●												□	
	●				●									□		
							●							□		
							●							□		
		●												□		
		●	●										●	□		
		●												□		
							●							□		
													●	□		
	●		●	●									●		□	
				●										□		
	●													□		
												●				□
				●			●							□		
											●					□

>>>

TOC

TABLE 2.1 Overview of form-finding parameters, performance objectives and topic for each reviewed paper

			Form-finding parameters																
			Adjacency matrix	Number of buildings	Building's shape	Ceiling design	Façade design	Floor height	Light shelf	Orientation	Roof's shape	Roof's structure	Shading	Space dimensions	Space location	2D / 3D grid matrix	Window dimensions	Window location	Window-to-wall ratio
Ref.	Author(s)	Year	x1	x2	x3	x4	x5	x6	x7	x8	x9	x10	x11	x12	x13	x14	x15	x16	x17
[125]	Rodrigues et al.	2013	x											x	x		x		
[75]	Talbourdet et al.	2013								x			x						x
[98]	Antonio et al.	2014		x			x								x	x			
[57]	Gerber and Lin	2014			x														
[119]	Karatas and El-Rayes	2014													x			x	x
[67]	Kasinalis et al.	2014																	x
[56]	Lin and Gerber	2014			x														
[55]	Lin and Gerber	2014			x														
[79]	Rafiq and Rustell	2014			x														x
[41]	R-González and Coley	2014											x	x					x
[76]	Wright et al.	2014											x						
[64]	Yi	2014			x									x					
[91]	Bizjak et al.	2015								x	x								
[77]	Brownlee and Wright	2015								x									x
[80]	Chang and Shih	2015			x					x									x
[111]	Ercan and Elias-Ozkan	2015					x						x						
[122]	Evins	2015											x				x		
[60]	Futrell et al.	2015				x			x				x				x		
[88]	Futrell et al.	2015				x							x						x
[105]	Huang et al.	2015			x											x			
[108]	Liu et al.	2015								x			x						x
[106]	McKinstray et al.	2015									x								x
[43]	Méndez Echenagucia et al.	2015															x		
[42]	Naboni et al.	2015								x			x						x
[61]	Negendahl and Nielsen	2015					x												
[90]	Su and Yan	2015								x				x	x		x		
[100]	Vermeulen et al.	2015			x														
[44]	Xu et al.	2015				x				x									x
[99]	Yi and Kim	2015			x														
[68]	Yu et al.	2015								x									x
[65]	Azari et al.	2016																	x
[89]	Chen et al.	2016								x			x						
[132]	Cubukcuoglu et al.	2016												x	x				

	Performance objectives													Topic		
	Energy	Daylight	Solar radiation	Environmental impact	Thermal comfort	Holistic sustainability	Life-cycle cost	Initial and global costs	Energy use cost	Holistic cost	Space allocation	Logistics	Structure	Building skin	Building shape	Building layout
	y1	y2	y3	y4	y5	y6	y7	y8	y9	y10	y11	y12	y13	z1	z2	z3
											●					□
	●							●						□		
			●													□
	●						●								□	
							●							□		
	●				●									□		
	●						●								□	
	●						●								□	
	●							●							□	
	●													□		
	●							●						□		
	●		●												□	
			●											□		
	●							●						□		
	●	●						●							□	
						●								□		
										●				□		
	●	●												□		
		●												□		
				●											□	
				●			●							□		
				●											□	
	●													□		
	●													□		
	●	●						●						□		
		●										●				□
			●													□
	●													□		
			●													□
	●				●									□		
	●			●										□		
		●			●									□		
												●				□

>>>

TABLE 2.1 Overview of form-finding parameters, performance objectives and topic for each reviewed paper

			Form-finding parameters																
			Adjacency matrix	Number of buildings	Building's shape	Ceiling design	Façade design	Floor height	Light shelf	Orientation	Roof's shape	Roof's structure	Shading	Space dimensions	Space location	2D / 3D grid matrix	Window dimensions	Window location	Window-to-wall ratio
Ref.	Author(s)	Year	x1	x2	x3	x4	x5	x6	x7	x8	x9	x10	x11	x12	x13	x14	x15	x16	x17
[46]	Delgarm et al.	2016								x			x				x		
[47]	Delgarm et al.	2016								x			x				x		
[131]	Dino	2016														x			
[120]	Ferrara et al.	2016															x		
[48]	Si et al.	2016								x								x	
[126]	Song et al.	2016			x									x	x				
[45]	Wright and Alajmi	2016					x												
[127]	Yazici	2016														x			
[96]	Zhang et al.	2016					x												
[133]	Bahrehmand et al.	2017	x												x				
[51]	Bamdad et al.	2017								x			x				x		
[50]	Bre and Fachinotti	2017								x			x				x		
[63]	Chatzikonstantinou et al.	2017											x				x		
[52]	Chen and Yang	2017								x			x						
[72]	Dhariwal and Banerjee	2017																	x
[115]	Dino and Üçoluk	2017														x			x
[135]	Elshaer et al.	2017			x														
[49]	Li et al.	2017								x							x		x
[114]	Sleiman et al.	2017													x				x
[78]	Yang et al.	2017											x				x		
[69]	Zhang et al.	2017								x			x	x					x
[53]	Bamdad et al.	2018											x						
[59]	Bizjak et al.	2018			x					x									x
[62]	Chen et al.	2018			x														x
[112]	Ferrara et al.	2018											x				x		
[71]	Gou et al.	2018								x			x						x
[73]	Harkouss et al.	2018															x		
[109]	Hester et al.	2018																	x
[58]	Li et al.	2018			x									x					x
[70]	Lin et al.	2018																	x
[110]	Sghiouri et al.	2018											x						
[97]	Vermeulen et al.	2018			x											x			
[84]	Yang et al.	2018									x	x	x						
[113]	Youssef et al.	2018			x		x												

	Performance objectives													Topic		
	Energy	Daylight	Solar radiation	Environmental impact	Thermal comfort	Holistic sustainability	Life-cycle cost	Initial and global costs	Energy use cost	Holistic cost	Space allocation	Logistics	Structure	Building skin	Building shape	Building layout
	y1	y2	y3	y4	y5	y6	y7	y8	y9	y10	y11	y12	y13	z1	z2	z3
	●													□		
	●													□		
												●				□
							●							□		
	●													□		
											●					□
	●													□		
											●					□
			●												□	
												●				□
	●													□		
	●													□		
	●	●												□		
	●													□		
	●						●							□		
						●								□		□
													●		□	
	●			●	●		●							□		
						●								□		□
	●							●						□		
	●	●			●									□		
	●													□		
	●														□	
	●	●													□	
						●								□		
	●				●									□		
	●						●							□		
				●			●							□		
	●														□	
	●				●									□		
					●									□		
			●												□	
	●	●											●	□		
						●								□	□	

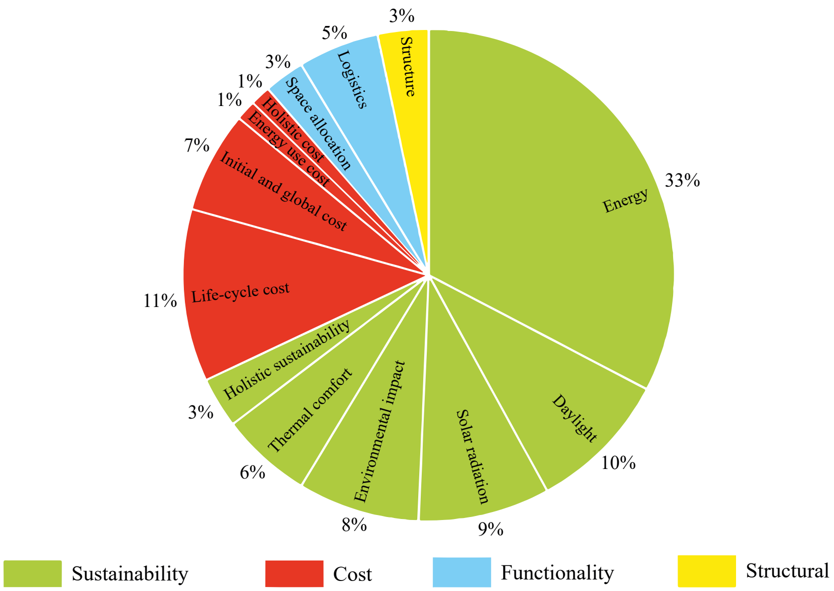


FIG. 2.4 Distribution of sub-categories

FIG. 2.5 Relationship between form-finding parameters and performance objectives

TABLE 2.2 Overview of optimisation methods and list of all parameters

Performance objectives	SEC methods used in PCA problems			All used parameters
	Layout	Skin	Shape	■ Related with form-finding parameters ■ Non-related with form-finding parameters
Energy	EC	EC&SI	EC	Window dimension, WWR, shading and ceiling design, building and roof shape, orientation, space dimension and location, light shelf, floor height, setpoints, temperature, construction and glazing properties, photovoltaic system, infiltration rate, solar absorptance, HVAC system and control variables
Daylight	EC	EC&SI	EC	Window dimension, WWR, shading and ceiling design, façade-roof-building shape, orientation, space dimension and location, light shelf, floor height, roof structure, window locations, construction and glazing properties, infiltration rate, HVAC system and control variables
Solar radiation	EC	EC	EC&SI	WWR, shading design, façade-roof-building shape, orientation, space dimension and location, roof structure, building urban layout, floor height
Environmental impact		EC&SI	EC&SI	Window dimension, WWR, shading design, building shape, orientation, setpoints, construction and glazing properties, building structure, air leakage, heat generator
Thermal comfort		EC&SI		Window dimension, WWR, shading design, building shape, orientation, setpoints, temperature, construction and glazing properties, solar absorptance, start-stop delays, relative humidity, airflow rate
Holistic sustainability	EC&SI	EC		WWR, shading design, façade and building shape, construction and glazing properties, space location, photovoltaic design, 2D/3D grid matrix for layout
Life-cycle cost		EC&SI	EC&SI	Window dimension, WWR, shading design, building shape, orientation, setpoints, construction and glazing properties, infiltration rate, HVAC system and control variables, humidity, ventilation, photovoltaic design, air leakage, heat generator
Initial and global cost	EC	EC	EC	Window dimension, WWR, façade and shading design, building and roof shapes, orientation, space dimension and location, floor count and height, construction and glazing properties, HVAC system
Energy use cost	EC			Window dimension, space dimension and location
Holistic cost		EC	EC	WWR, shading design, structural and floor system, construction and glazing properties, renewables, plant and storage
Space allocation	EC			Building shape, space dimension and location, geometric transformation, grid subdivision
Logistics	EC			Window dimension, orientation, space dimension and location, facility assignment, adjacency matrix and preference, space properties, voxel matrix, window-door-entrance placement
Structure		EC	EC	Building and façade shape, roof structure, geometric transformation

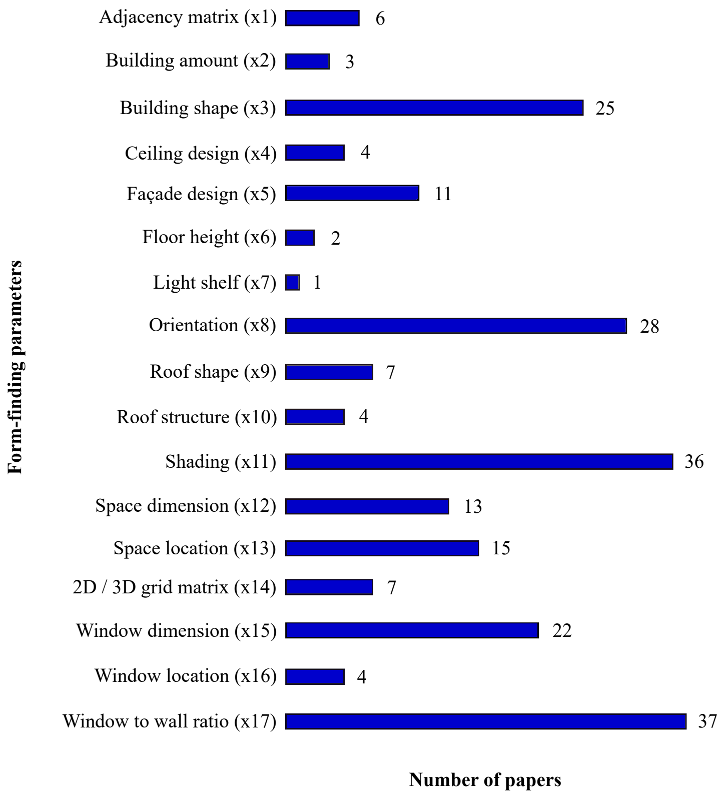


FIG. 2.6 Total number of form-finding parameters

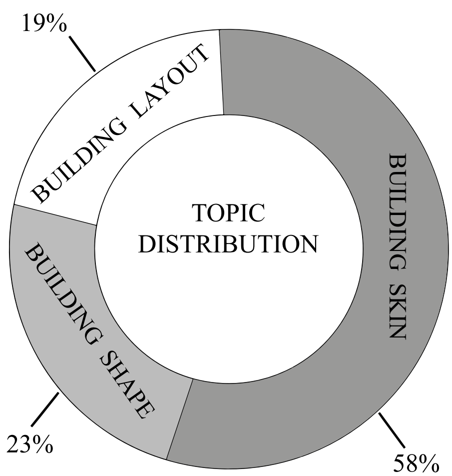


FIG. 2.7 Distribution of building optimisation topics

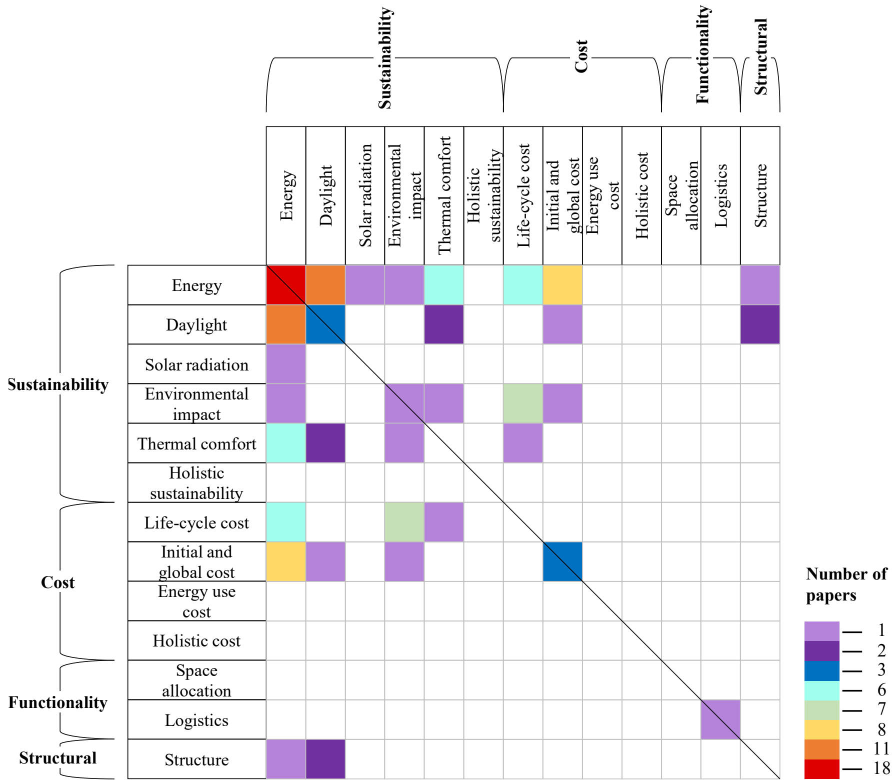


FIG. 2.8 Matrix showing the use of conflicting objectives

Evaluating of these results allows the extraction of some information as follows:

- From the viewpoint of the main categories, Fig. 2.4 shows that sustainability was the most studied topic. The least studied category was structure
- From the viewpoint sub-categories, energy was the most dominant performance objective among all reviewed papers as it can be observed in Fig. 2.4, while little attention was attributed to holistic and energy use costs. The efforts on other sub-categories were almost equally distributed
- As in Table 2.2, EC was the major optimisation method used in reviewed papers

TOC

- As shown in Table 2.2, energy and daylight related papers considered a broader range of parameters than other performance topics. In addition to form-finding related parameters, non-related form-finding parameters played a crucial role in energy, daylight, environmental impact, thermal comfort, life-cycle cost, initial and global costs, and holistic cost considerations

- From the viewpoint of form-finding parameters illustrated in Fig. 2.6, WWR (x17), shading (x11), orientation (x8), window dimensions (x15), and building's shape (x3), were mostly used. Conversely, number of buildings (x2), ceiling design (x4), floor height (x6), light shelf (x7), roof's structure (x10), and window location (x16), were the least used form-finding parameters

- By matching the information of Fig. 2.6 and Table 2.1, the relationship between mostly used form-finding parameters and the corresponding building topic was investigated in detail. The window-to-wall ratio (x17) was used 28 times in relation to the building's skin from a total of 37 cases. The remaining cases were related to building's shape. The window dimensions (x15) was used 17 times (out of 22 cases in total) in relation to building's skin. Moreover, in one occasion (from a total of 22 times) it was used in relation to the building's shape, whereas in four occasions (out of 22 total cases) it was used in the building's layout. Shading (x11) was used 33 times (out of 36 cases in total) in relation to the building's skin. In addition, shading was studied in relation to the building's shape in the remaining cases. Orientation (x8) was used 22 times (out of 28 in total) in relation to the building's skin whereas in a building layout it was used in one case (out of 22 times). The rest of the times it was studied in relation to the building's shape. Finally, building's shape (x3) was used two times (from a total of 25 cases) in relation to the use of the building's skin. Furthermore, in 20 out of 25 times it was used in the building's shape whereas in three cases (out of 25 times) it was used in a building layout

- As shown in Fig. 2.7, the major building topic was the building's skin. Studies on the building layout and building shape were almost equally distributed

- As shown in Fig. 2.8, combinations of objectives in regard to sustainability were dominant. In contrast, the second most dominant combination was between sustainability and cost. Among the sub-categories, combinations of these objectives that related to either energy or energy and daylight were the most common trade-offs. Trade-offs between functionality and sustainability were obviously neglected in multi-objective approaches, though they were only considered in weighted sum approaches, e.g. in [90]. Furthermore, the trade-offs between functionality and cost were neglected both in the multi-objective and in weighted sum approaches

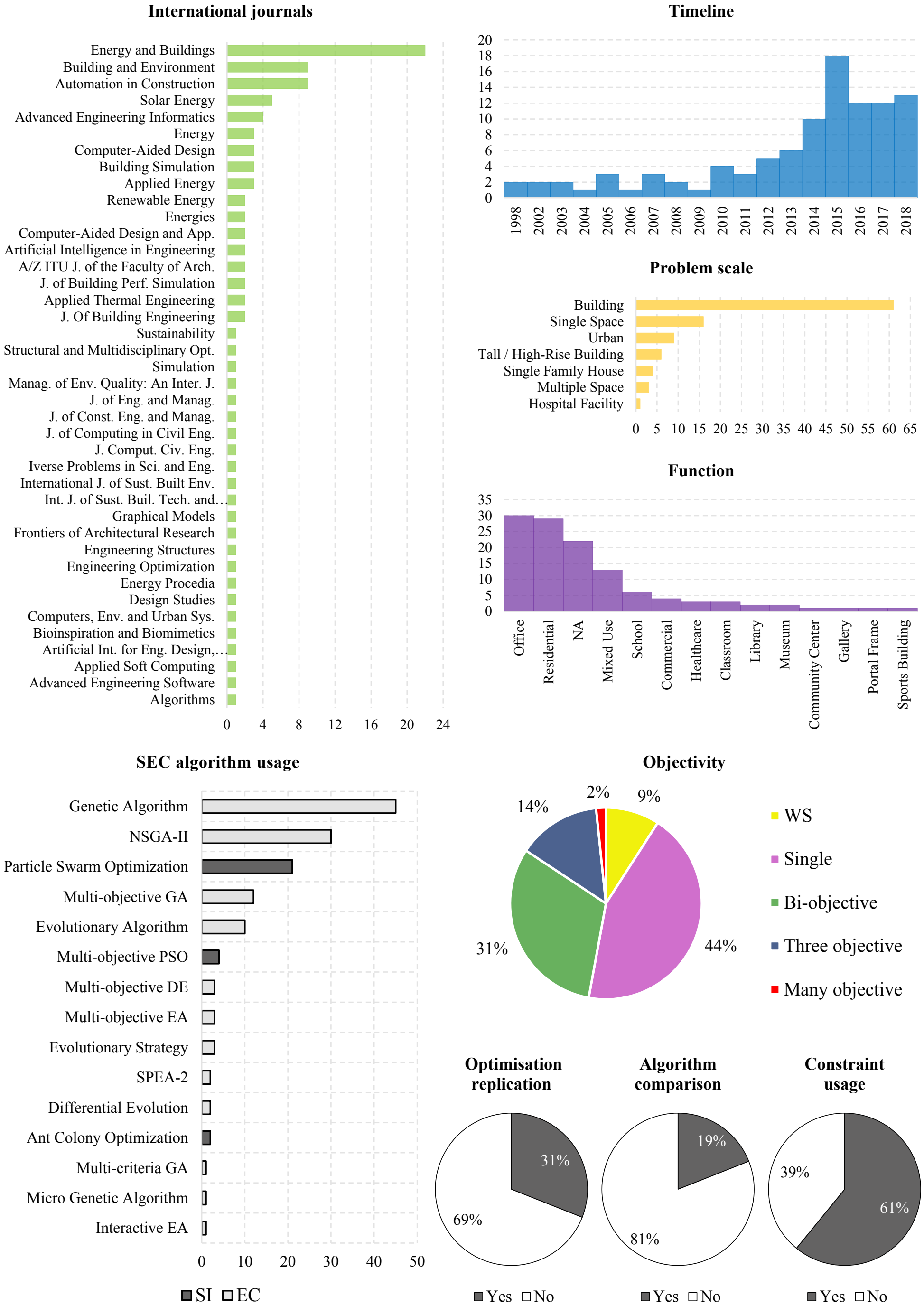


FIG. 2.9 Current trends

TOC

- As shown in Fig. 2.9, the number of published papers increased significantly in the last five years. Most of the reviewed papers were published in “Energy and Buildings”, “Building and Environment”, and “Automation in Construction”. The most studied problem scale was the building and the functions that received maximum attention were offices and residences

- Single-objective and bi-objective optimisation problems were mostly considered, second to weighted summation (WS) and three objective optimisation problems. Many objective optimisation problems have rarely been considered. Most of the reviewed studies used constraints in the formulation of the optimisation problem

- In approximately one-third of the published studies, optimisation replications were considered (e.g. different initial populations were used for each of the runs described within one publication). In addition, comparisons of the optimisation results using several SEC algorithms were also very limited

- GA was mostly used for single-objective optimisation, whereas NSGA-II was mostly used for multi-objective optimisation problems

2.8 Conclusion

This study provides a systematic review of PCA using SEC. The topic has been on the agenda of architects and engineers during the past few decades. Based on the evidence presented in the results, conclusions were drawn in relation to form-finding parameters, performance objectives, and optimisation, as summarised below.

Conclusions on form-finding parameters

- All reviewed publications dealt with the architectural design (this is because they all included form-finding as explained in the introduction of this study). Nevertheless, some of these publications placed more emphasis on architectural concerns (and therefore include parameters that have great impact on architectural design, such building’s shape (x3)). These publications are also the ones that focussed their conclusions on the building design. Other reviewed publications placed more emphasis on engineering concerns (such as the window-to-wall ratio (x17)). Several of these publications are also the ones that focus their conclusions on aspects related to computer science (such as algorithmic comparisons). A better integration of investigations related to computer science within the architectural domain is missing despite its expected benefits

- Form-finding parameters, such as window-to-wall ratio (x17) and shading (x11) can enhance sustainability performances. However, these parameters should be more representative compared to the window ratio or shading dimensions by including more design concerns

- Based on the sustainability objective in building layout problems [81,90], authors tended to use window dimensions (x15) instead of the window-to-wall ratio (x17). One reason is the fact that the window dimension parameters are more controllable in relation to variations in the layout

- Aspects that are usually delegated to shading (x11) (such as the control of solar gain for thermal comfort, control of the amount of daylight for visual comfort and prevention of glare) can also be improved by layout optimisation. Currently, this potential is not exploited given the lack of works that use shading parameters (x11) in the layout topic

- Orientation (x8) is one of the most crucial parameters used to improve the sustainability and functionality performances of the building. From this point of view, it is remarkable that the orientation (x8) parameter has been rarely investigated in relation to the building layout. Currently, potentials are not exploited when the orientation parameter (x8) is incorporated in layout problems

- There is no doubt that the façade design affects the sustainability-related performance objectives. In the reviewed publications, orientation parameters (x8) are mostly used in building skin problems. However, orientation can also be controlled by the variations of the building shape (e.g. twisted building)

- Among the least used parameters, the light shelf (x7) has only been used in only one study thus far [60]. In order to improve the sustainability performance of existing buildings, the light shelf related parameters can play an important role

Conclusions on performance objectives

- Only three studies [61,73,84] solved many-objective optimisation problems that involved all four objectives. Considering the necessity of satisfying many aspects of the design process, many-objective optimisation algorithms could be used.

When doing so, the selection of the optimisation algorithm is crucial since some of these are not convenient for use in many-objective optimisation problems. In the literature, there are some novel and recent techniques, which can be found in [136]

- There were only three holistic approaches [113-115] that focussed on different building topics as part of the same optimisation problem, such as the skin and layout. These approaches presented promising potentials, while they integrated different performance objectives and minimised the design complexity. These approaches can be considered, especially in research studies that are focussed on integrated design approaches for high-performance buildings

- Owing to the expensive computation time, the number of large-scale building studies, such as tall buildings and hospitals, was limited. Objective functions based on ANN can be an effective solution

Conclusions on optimisation

- There are several methods to handle more than one objective in the multi-objective optimisation problem. One of them is the weighted summation approach which appears to be relevant and used in many of the reviewed documents. It combines different objectives by assigning some weights to each objective in order to convert the problem to a single-objective optimisation problem. However, defining these weights is a very difficult task (especially if this needs to be done at the beginning of the process, such as the case of weighted summation). Conversely, Pareto-optimality approaches in multi-objective problems (e.g. bi-objective or three objective problems) allow the identification of the final decision at the end of the optimisation process

- Owing to the formulation of optimisation problem, the final design decision can consider and eventually incorporate the results of the optimisation in different ways. From the standpoint of the single-objective problem, the result of the optimisation process can be used as the final design decision. The reason is that in this approach, there is only one fitness function to be either minimised or maximised. From the standpoint of the weighted summation problem, the final design decision depends on the weights that are defined for each fitness function. Even if there is only one result at the end of this optimisation process, the weights (defined by the decision-maker) may affect the result. From the standpoint of multi-objective problems, further investigation is required after the optimisation. The reason is that the Pareto-front suggests many alternatives as result. Owing to the non-domination, each alternative has either an advantage or disadvantage for each objective function based on where the alternative is selected from the Pareto-front. Considering multi-objective optimisation may support the investigation of the relationship among objectives and decision variables. In reviewed documents, there are several methods explicitly used to support the decision-making process in the multi-objective domain (e.g. weighted summation approach to pick the closest solution to the utopic point in [46,47], clustering to categorise solutions in Pareto-front using self-organising maps in [84], and auto-associative connectionist model for treating preferences in [63]). A further consideration is valid for all types of optimisations and relates to the complexity of the architectural design. Regardless of how many objectives it can include, the optimisation tackles only a limited range of design requirements. Many other requirements and qualities expected from the design are not included in the optimisation. The optimisation results may have to be assessed and further elaborated also based on these additional criteria

- According to the “No Free Lunch (NFL)” theorem [137], there is no global metaheuristics optimisation algorithm that is capable of discovering the best results for all real-world or benchmark problems. In other words, one algorithm can outperform another algorithm only in terms of the solution to a specific problem. Architectural designs are unique problems owing to objectives, building programs, constraints, client expectations, and the surrounding impacts of the built environment. Therefore, one must explore and compare different algorithms for solving the same architectural design problem in order to eventually provide more adequate design decisions. However, in the literature, it is observed that very few studies have compared different SEC algorithms for the same architectural design problem

- Only one study [97] explicitly considered equality constraints, which are very important in architecture (e.g. in order to match the design with strict municipality regulations). The use of equality constraint causes is associated with a more challenging optimisation process while searching for feasible design solutions. For this reason, only specialised constraint handling methods can cope with equality constraint problems [138]

- Many objective optimisation problems are challenging in terms of observing the final set of solutions in the search space since they integrate at least four objectives at the same time. Different visualisation methods should be considered in order to facilitate the design choice

Given the tremendous effort expended on research on this topic, the relevance of the PCA framework is confirmed by the conducted review of the one hundred articles considered and referenced in this study. Nevertheless, the itemised points listed above clearly indicate the directions in which further efforts are needed.

References

[1] Shi, X.; Yang, W., Performance-driven architectural design and optimization technique from a perspective of architects. *Automation in Construction* **2013,** 32, 125-135.

[2] Cobb, P.; Confrey, J.; DiSessa, A.; Lehrer, R.; Schauble, L., Design experiments in educational research. *Educational researcher* **2003,** 32, (1), 9-13.

[3] Kolarevic, B. In *Computing the performative in architecture*, Proceedings of the 21th eCAADe Conference: Digital Design. Graz, Austria, **2003**; pp 17-20.

[4] Sariyildiz, I. In *Performative computational design*, Keynote speech in: Proceedings of ICONARCH-I: International congress of architecture-I, Konya, Turkey, 15-17 November **2012**, Selcuk University.

[5] Huang, Y.; Niu, J. L., Optimal building envelope design based on simulated performance: History, current status and new potentials. *Energy and Buildings* **2016,** 117, 387-398.

[6] Machairas, V.; Tsangrassoulis, A.; Axarli, K., Algorithms for optimization of building design: A review. *Renewable and Sustainable Energy Reviews* **2014,** 31, 101-112.

[7] Michalewicz, Z.; Fogel, D. B., *How to solve it: modern heuristics*. Springer Science & Business Media: **2013**.

[8] Blum, C.; Li, X., Swarm intelligence in optimization. In *Swarm Intelligence*, Springer: **2008**; pp 43-85.

[9] Mitchell, M.; Taylor, C. E., Evolutionary computation: an overview. *Annual Review of Ecology and Systematics* **1999,** 30, (1), 593-616.

[10] Radford, A. D.; Gero, J. S., On optimization in computer aided architectural design. *Building and Environment* **1980,** 15, (2), 73-80.

[11] Touloupaki, E.; Theodosiou, T., Performance Simulation Integrated in Parametric 3D Modeling as a Method for Early Stage Design Optimization—A Review. *Energies* **2017,** 10, (5), 637.

[12] Negendahl, K., Building performance simulation in the early design stage: An introduction to integrated dynamic models. *Automation in Construction* **2015,** 54, 39-53.

[13] Nguyen, A.-T.; Reiter, S.; Rigo, P., A review on simulation-based optimization methods applied to building performance analysis. *Applied Energy* **2014,** 113, 1043-1058.

[14] Dutta, K.; Sarthak, S., Architectural space planning using evolutionary computing approaches: A review. *Artificial Intelligence Review* **2011,** 36, (4), 311-321.

[15] Evins, R., A review of computational optimisation methods applied to sustainable building design. *Renewable & Sustainable Energy Reviews* **2013,** 22, 230-245.

[16] Attia, S.; Hamdy, M.; O'Brien, W.; Carlucci, S., Assessing gaps and needs for integrating building performance optimization tools in net zero energy buildings design. *Energy and Buildings* **2013,** 60, 110-124.

[17] Shi, Z.; Fonseca, J. A.; Schlueter, A., A review of simulation-based urban form generation and optimization for energy-driven urban design. *Building and Environment* **2017,** 121, 119-129.

[18] Cui, Y.; Geng, Z.; Zhu, Q.; Han, Y., Multi-objective optimization methods and application in energy saving. *Energy* **2017,** 125, 681-704.

[19] Tian, Z.; Zhang, X.; Jin, X.; Zhou, X.; Si, B.; Shi, X., Towards adoption of building energy simulation and optimization for passive building design: a survey and a review. *Energy and Buildings* **2017**.

[20] Kheiri, F., A review on optimization methods applied in energy-efficient building geometry and envelope design. *Renewable and Sustainable Energy Reviews* **2018,** 92, 897-920.

[21] Shi, X.; Tian, Z.; Chen, W.; Si, B.; Jin, X., A review on building energy efficient design optimization rom the perspective of architects. *Renewable and Sustainable Energy Reviews* **2016,** 65, 872-884.

[22] Eltaweel, A.; Yuehong, S., Parametric design and daylighting: A literature review. *Renewable and Sustainable Energy Reviews* **2017,** 73, 1086-1103.

[23] Veenendaal, D.; Block, P., An overview and comparison of structural form finding methods for general networks. *International Journal of Solids and Structures* **2012,** 49, (26), 3741-3753.

[24] Adriaenssens, S.; Block, P.; Veenendaal, D.; Williams, C., *Shell structures for architecture: form finding and optimization*. Routledge: **2014**.

[25] Haber, R.; Abel, J., Initial equilibrium solution methods for cable reinforced membranes part I—formulations. *Computer Methods in Applied Mechanics and Engineering* **1982,** 30, (3), 263-284.

[26] Lewis, W. J., *Tension structures: form and behaviour*. Thomas Telford: **2003**.

[27] Broadbent, G., Design method in architecture. *Design Methods in Architecture, Lund Humphries, London* **1969**.

[28] Hubka, V.; Eder, W. E., A scientific approach to engineering design. *Design studies* **1987,** 8, (3), 123-137.

[29] Turrin, M., *Performance Assessment Strategies: A computational framework for conceptual design of large roofs*. TU Delft: **2013**.
[30] Dino, I., Creative design exploration by parametric generative systems in architecture. *METU Journal of Faculty of Architecture* **2012,** 29, (1), 207-224.
[31] Ciftcioglu, O.; Sariyildiz, I. S.; Bittermann, M. S. In *Building performance analysis supported by GA*, Evolutionary Computation, CEC IEEE Congress on: **2007**; pp 859-866.
[32] Holland, J.; Goldberg, D., Genetic algorithms in search, optimization and machine learning. *Massachusetts: Addison-Wesley* **1989**.
[33] Storn, R.; Price, K., Differential evolution–a simple and efficient heuristic for global optimization over continuous spaces. *Journal of global optimization* **1997,** 11, (4), 341-359.
[34] Eberhart, R.; Kennedy, J. In *A new optimizer using particle swarm theory*, Micro Machine and Human Science, MHS'95., Proceedings of the Sixth International Symposium on, IEEE: **1995**; pp 39-43.
[35] Dorigo, M.; Birattari, M.; Stutzle, T., Ant colony optimization. *IEEE computational intelligence magazine* **2006,** 1, (4), 28-39.
[36] Caldas, L. G.; Norford, L. K., A design optimization tool based on a genetic algorithm. *Automation in Construction* **2002,** 11, (2), 173-184.
[37] Caldas, L.; Norford, L.; Rocha, J., An evolutionary model for sustainable design. *Management of Environmental Quality: An International Journal* **2003,** 14, (3), 383-397.
[38] Wetter, M.; Wright, J., A comparison of deterministic and probabilistic optimization algorithms for nonsmooth simulation-based optimization. *Building and Environment* **2004,** 39, (8), 989-999.
[39] Lee, J. H., Optimization of indoor climate conditioning with passive and active methods using GA and CFD. *Building and Environment* **2007,** 42, (9), 3333-3340.
[40] Bucking, S.; Zmeureanu, R.; Athienitis, A., An information driven hybrid evolutionary algorithm for optimal design of a Net Zero Energy House. *Solar Energy* **2013,** 96, 128-139.
[41] Ramallo-González, A. P.; Coley, D. A., Using self-adaptive optimisation methods to perform sequential optimisation for low-energy building design. *Energy and Buildings* **2014,** 81, 18-29.
[42] Naboni, E.; Malcangi, A.; Zhang, Y.; Barzon, F., Defining The Energy Saving Potential of Architectural Design. *Energy Procedia* **2015,** 83, 140-146.
[43] Méndez Echenagucia, T.; Capozzoli, A.; Cascone, Y.; Sassone, M., The early design stage of a building envelope: Multi-objective search through heating, cooling and lighting energy performance analysis. *Applied Energy* **2015,** 154, 577-591.
[44] Xu, J.; Kim, J. H.; Hong, H.; Koo, J., A systematic approach for energy efficient building design factors optimization. *Energy and Buildings* **2015,** 89, 87-96.
[45] Wright, J.; Alajmi, A., Efficient Genetic Algorithm sets for optimizing constrained building design problem. *International Journal of Sustainable Built Environment* **2016,** 5, (1), 123-131.
[46] Delgarm, N.; Sajadi, B.; Kowsary, F.; Delgarm, S., Multi-objective optimization of the building energy performance: A simulation-based approach by means of particle swarm optimization (PSO). *Applied Energy* **2016,** 170, 293-303.
[47] Delgarm, N.; Sajadi, B.; Delgarm, S.; Kowsary, F., A novel approach for the simulation-based optimization of the buildings energy consumption using NSGA-II: Case study in Iran. *Energy and Buildings* **2016,** 127, 552-560.
[48] Si, B.; Tian, Z.; Jin, X.; Zhou, X.; Tang, P.; Shi, X., Performance indices and evaluation of algorithms in building energy efficient design optimization. *Energy* **2016,** 114, 100-112.
[49] Li, K. J.; Pan, L.; Xue, W. P.; Jiang, H.; Mao, H. P., Multi-Objective Optimization for Energy Performance Improvement of Residential Buildings: A Comparative Study. *Energies* **2017,** 10, (2).
[50] Bre, F.; Fachinotti, V. D., A computational multi-objective optimization method to improve energy efficiency and thermal comfort in dwellings. *Energy and Buildings* **2017,** 154, 283-294.
[51] Bamdad, K.; Cholette, M. E.; Guan, L.; Bell, J., Ant colony algorithm for building energy optimisation problems and comparison with benchmark algorithms. *Energy and Buildings* **2017,** 154, 404-414.
[52] Chen, X.; Yang, H., A multi-stage optimization of passively designed high-rise residential buildings in multiple building operation scenarios. *Applied Energy* **2017,** 206, 541-557.
[53] Bamdad, K.; Cholette, M. E.; Guan, L.; Bell, J., Building energy optimisation under uncertainty using ACOMV algorithm. *Energy and Buildings* **2018,** 167, 322-333.
[54] Caldas, L., Generation of energy-efficient architecture solutions applying GENE_ARCH: An evolution-based generative design system. *Advanced Engineering Informatics* **2008,** 22, (1), 59-70.

[55] Lin, S. H.; Gerber, D. J., Evolutionary energy performance feedback for design: Multidisciplinary design optimization and performance boundaries for design decision support. *Energy and Buildings* **2014,** 84, 426-441.
[56] Lin, S. H. E.; Gerber, D. J., Designing-in performance: A framework for evolutionary energy performance feedback in early stage design. *Automation in Construction* **2014,** 38, 59-73.
[57] Gerber, D. J.; Lin, S. H. E., Designing in complexity: Simulation, integration, and multidisciplinary design optimization for architecture. *Simulation-Transactions of the Society for Modeling and Simulation International* **2014,** 90, (8), 936-959.
[58] Li, Z.; Chen, H.; Lin, B.; Zhu, Y. In *Fast bidirectional building performance optimization at the early design stage*, Building Simulation, Springer: **2018**; pp 647-661.
[59] Bizjak, M.; Žalik, B.; Štumberger, G.; Lukač, N., Estimation and optimisation of buildings' thermal load using LiDAR data. *Building and Environment* **2018,** 128, 12-21.
[60] Futrell, B. J.; Ozelkan, E. C.; Brentrup, D., Bi-objective optimization of building enclosure design for thermal and lighting performance. *Building and Environment* **2015,** 92, 591-602.
[61] Negendahl, K.; Nielsen, T. R., Building energy optimization in the early design stages: A simplified method. *Energy and Buildings* **2015,** 105, 88-99.
[62] Chen, K. W.; Janssen, P.; Schlueter, A., Multi-objective optimisation of building form, envelope and cooling system for improved building energy performance. *Automation in Construction* **2018,** 94, 449-457.
[63] Chatzikonstantinou, I.; Sariyildiz, I. S., Addressing design preferences via auto-associative connectionist models: Application in sustainable architectural Façade design. *Automation in Construction* **2017,** 83, 108-120.
[64] Yi, H., Automated generation of optimised building envelope: simulation based multi-objective process using evolutionary algorithm. *International Journal of Sustainable Building Technology and Urban Development* **2014,** 5, (3), 159-170.
[65] Azari, R.; Garshasbi, S.; Amini, P.; Rashed-Ali, H.; Mohammadi, Y., Multi-objective optimization of building envelope design for life cycle environmental performance. *Energy and Buildings* **2016,** 126, 524-534.
[66] Magnier, L.; Haghighat, F., Multiobjective optimization of building design using TRNSYS simulations, genetic algorithm, and Artificial Neural Network. *Building and Environment* **2010,** 45, (3), 739-746.
[67] Kasinalis, C.; Loonen, R. C. G. M.; Cóstola, D.; Hensen, J. L. M., Framework for assessing the performance potential of seasonally adaptable facades using multi-objective optimization. *Energy and Buildings* **2014,** 79, 106-113.
[68] Yu, W.; Li, B.; Jia, H.; Zhang, M.; Wang, D., Application of multi-objective genetic algorithm to optimize energy efficiency and thermal comfort in building design. *Energy and Buildings* **2015,** 88, 135-143.
[69] Zhang, A.; Bokel, R.; van den Dobbelsteen, A.; Sun, Y.; Huang, Q.; Zhang, Q., Optimization of thermal and daylight performance of school buildings based on a multi-objective genetic algorithm in the cold climate of China. *Energy and Buildings* **2017,** 139, 371-384.
[70] Lin, Y.; Zhou, S.; Yang, W.; Li, C.-Q., Design Optimization Considering Variable Thermal Mass, Insulation, Absorptance of Solar Radiation, and Glazing Ratio Using a Prediction Model and Genetic Algorithm. *Sustainability* **2018,** 10, (2), 336.
[71] Gou, S.; Nik, V. M.; Scartezzini, J.-L.; Zhao, Q.; Li, Z., Passive design optimization of newly-built residential buildings in Shanghai for improving indoor thermal comfort while reducing building energy demand. *Energy and Buildings* **2018,** 169, 484-506.
[72] Dhariwal, J.; Banerjee, R., An approach for building design optimization using design of experiments. *Building Simulation* **2017,** 10, (3), 323-336.
[73] Harkouss, F.; Fardoun, F.; Biwole, P. H., Multi-objective optimization methodology for net zero energy buildings. *Journal of Building Engineering* **2018,** 16, 57-71.
[74] Znouda, E.; Ghrab-Morcos, N.; Hadj-Alouane, A., Optimization of Mediterranean building design using genetic algorithms. *Energy and Buildings* **2007,** 39, (2), 148-153.
[75] Talbourdet, F.; Michel, P.; Andrieux, F.; Millet, J. R.; Mankibi, M. E.; Vinot, B., A knowledge-aid approach for designing high-performance buildings. *Building Simulation* **2013,** 6, (4), 337-350.
[76] Wright, J. A.; Brownlee, A.; Mourshed, M. M.; Wang, M., Multi-objective optimization of cellular fenestration by an evolutionary algorithm. *Journal of Building Performance Simulation* **2014,** 7, (1), 33-51.
[77] Brownlee, A. E. I.; Wright, J. A., Constrained, mixed-integer and multi-objective optimisation of building designs by NSGA-II with fitness approximation. *Applied Soft Computing Journal* **2015,** 33, 114-126.

[78] Yang, M. D.; Lin, M. D.; Lin, Y. H.; Tsai, K. T., Multiobjective optimization design of green building envelope material using a non-dominated sorting genetic algorithm. *Applied Thermal Engineering* **2017,** 111, 1255-1264.
[79] Rafiq, M. Y.; Rustell, M. J., Building information modeling steered by evolutionary computing. *Journal of Computing in Civil Engineering* **2014,** 28, (4).
[80] Chang, M. C.; Shih, S. G., A Hybrid Approach of Dynamic Programming and Genetic Algorithm for Multi-criteria Optimization on Sustainable Architecture design. *Computer-Aided Design and Applications* **2015,** 12, (3), 310-319.
[81] Michalek, J. J.; Choudhary, R.; Papalambros, P. Y., Architectural layout design optimization. *Engineering Optimization* **2002,** 34, (5), 461-484.
[82] Baušys, R.; Pankrašovaité, I., Optimization of architectural layout by the improved genetic algorithm. *Journal of Civil Engineering and Management* **2005,** 11, (1), 13-21.
[83] Menges, A., Biomimetic design processes in architecture: Morphogenetic and evolutionary computational design. *Bioinspiration and Biomimetics* **2012,** 7, (1).
[84] Yang, D.; Ren, S.; Turrin, M.; Sariyildiz, S.; Sun, Y., Multi-disciplinary and multi-objective optimization problem re-formulation in computational design exploration: A case of conceptual sports building design. *Automation in Construction* **2018,** 92, 242-269.
[85] Turrin, M.; Von Buelow, P.; Stouffs, R., Design explorations of performance driven geometry in architectural design using parametric modeling and genetic algorithms. *Advanced Engineering Informatics* **2011,** 25, (4), 656-675.
[86] Rakha, T.; Nassar, K., Genetic algorithms for ceiling form optimization in response to daylight levels. *Renewable Energy* **2011,** 36, (9), 2348-2356.
[87] Gagne, J.; Andersen, M., A generative facade design method based on daylighting performance goals. *Journal of Building Performance Simulation* **2012,** 5, (3), 141-154.
[88] Futrell, B. J.; Ozelkan, E. C.; Brentrup, D., Optimizing complex building design for annual daylighting performance and evaluation of optimization algorithms. *Energy and Buildings* **2015,** 92, 234-245.
[89] Chen, X.; Yang, H.; Sun, K., A holistic passive design approach to optimize indoor environmental quality of a typical residential building in Hong Kong. *Energy* **2016,** 113, 267-281.
[90] Su, Z. Z.; Yan, W., A fast genetic algorithm for solving architectural design optimization problems. *Ai Edam-Artificial Intelligence for Engineering Design Analysis and Manufacturing* **2015,** 29, (4), 457-469.
[91] Bizjak, M.; Žalik, B.; Lukač, N., Evolutionary-driven search for solar building models using LiDAR data. *Energy and Buildings* **2015,** 92, 195-203.
[92] Liu, X.; Liu, H.; Duan, H., Particle swarm optimization based on dynamic niche technology with applications to conceptual design. *Advances in Engineering Software* **2007,** 38, (10), 668-676.
[93] Oliveira Panão, M. J. N.; Gonçalves, H. J. P.; Ferrão, P. M. C., Optimization of the urban building efficiency potential for mid-latitude climates using a genetic algorithm approach. *Renewable Energy* **2008,** 33, (5), 887-896.
[94] Kämpf, J. H.; Robinson, D., Optimisation of building form for solar energy utilisation using constrained evolutionary algorithms. *Energy and Buildings* **2010,** 42, (6), 807-814.
[95] Kämpf, J. H.; Montavon, M.; Bunyesc, J.; Bolliger, R.; Robinson, D., Optimisation of buildings' solar irradiation availability. *Solar energy* **2010,** 84, (4), 596-603.
[96] Zhang, L.; Zhang, L.; Wang, Y., Shape optimization of free-form buildings based on solar radiation gain and space efficiency using a multi-objective genetic algorithm in the severe cold zones of China. *Solar Energy* **2016,** 132, 38-50.
[97] Vermeulen, T.; Merino, L.; Knopf-Lenoir, C.; Villon, P.; Beckers, B., Periodic urban models for optimization of passive solar irradiation. *Solar Energy* **2018,** 162, 67-77.
[98] António, C. A. C.; Monteiro, J. B.; Afonso, C. F., Optimal topology of urban buildings for maximization of annual solar irradiation availability using a genetic algorithm. *Applied Thermal Engineering* **2014,** 73, (1), 424-437.
[99] Yi, Y. K.; Kim, H., Agent-based geometry optimization with Genetic Algorithm (GA) for tall apartment's solar right. *Solar Energy* **2015,** 113, 236-250.
[100] Vermeulen, T.; Knopf-Lenoir, C.; Villon, P.; Beckers, B., Urban layout optimization framework to maximize direct solar irradiation. *Computers Environment and Urban Systems* **2015,** 51, 1-12.
[101] Wang, W.; Zmeureanu, R.; Rivard, H., Applying multi-objective genetic algorithms in green building design optimization. *Building and Environment* **2005,** 40, (11), 1512-1525.

[102] Rapone, G.; Saro, O., Optimisation of curtain wall facades for office buildings by means of PSO algorithm. *Energy and Buildings* **2012,** 45, 189-196.
[103] Wang, W.; Rivard, H.; Zmeureanu, R., An object-oriented framework for simulation-based green building design optimization with genetic algorithms. *Advanced Engineering Informatics* **2005,** 19, (1), 5-23.
[104] Wang, W.; Rivard, H.; Zmeureanu, R., Floor shape optimization for green building design. *Advanced Engineering Informatics* **2006,** 20, (4), 363-378.
[105] Huang, Y. S.; Chang, W. S.; Shih, S. G., Building Massing Optimization in the Conceptual Design Phase. *Computer-Aided Design and Applications* **2015,** 12, (3), 344-354.
[106] McKinstray, R.; Lim, J. B.; Tanyimboh, T. T.; Phan, D. T.; Sha, W.; Brownlee, A. E., Topographical optimisation of single-storey non-domestic steel framed buildings using photovoltaic panels for net-zero carbon impact. *Building and Environment* **2015,** 86, 120-131.
[107] Karatas, A.; El-Rayes, K., Optimizing tradeoffs among housing sustainability objectives. *Automation in Construction* **2013,** 53, 83-94.
[108] Liu, S.; Meng, X.; Tam, C., Building information modeling based building design optimization for sustainability. *Energy and Buildings* **2015,** 105, 139-153.
[109] Hester, J.; Gregory, J.; Ulm, F.-J.; Kirchain, R., Building design-space exploration through quasi-optimization of life cycle impacts and costs. *Building and Environment* **2018**.
[110] Sghiouri, H.; Mezrhab, A.; Karkri, M.; Naji, H., Shading devices optimization to enhance thermal comfort and energy performance of a residential building in Morocco. *Journal of Building Engineering* **2018,** 18, 292-302.
[111] Ercan, B.; Elias-Ozkan, S. T., Performance-based parametric design explorations: A method for generating appropriate building components. *Design Studies* **2015,** 38, 33-53.
[112] Ferrara, M.; Sirombo, E.; Fabrizio, E., Automated optimization for the integrated design process: the energy, thermal and visual comfort nexus. *Energy and Buildings* **2018,** 168, 413-427.
[113] Youssef, A. M.; Zhai, Z. J.; Reffat, R. M., Generating proper building envelopes for photovoltaics integration with shape grammar theory. *Energy and Buildings* **2018,** 158, 326-341.
[114] Sleiman, H. A.; Hempel, S.; Traversari, R.; Bruinenberg, S., An Assisted Workflow for the Early Design of Nearly Zero Emission Healthcare Buildings. *Energies* **2017,** 10, (7), 993.
[115] Dino, I. G.; Üçoluk, G., Multiobjective Design Optimization of Building Space Layout, Energy, and Daylighting Performance. *Journal of Computing in Civil Engineering* **2017,** 31, (5), 04017025.
[116] Tuhus-Dubrow, D.; Krarti, M., Genetic-algorithm based approach to optimize building envelope design for residential buildings. *Building and Environment* **2010,** 45, (7), 1574-1581.
[117] Bichiou, Y.; Krarti, M., Optimization of envelope and HVAC systems selection for residential buildings. *Energy and Buildings* **2011,** 43, (12), 3373-3382.
[118] Gengembre, E.; Ladevie, B.; Fudym, O.; Thuillier, A., A Kriging constrained efficient global optimization approach applied to low-energy building design problems. *Inverse Problems in Science and Engineering* **2012,** 20, (7), 1101-1114.
[119] Karatas, A.; El-Rayes, K., Optimal trade-offs between social quality of life and life-cycle cost in housing units. *Journal of Construction Engineering and Management* **2014,** 140, (12).
[120] Ferrara, M.; Fabrizio, E.; Virgone, J.; Filippi, M., Energy systems in cost-optimized design of nearly zero-energy buildings. *Automation in Construction* **2016,** 70, 109-127.
[121] Gero, J. S.; Kazakov, V. A., Evolving design genes in space layout planning problems. *Artificial Intelligence in Engineering* **1998,** 12, (3), 163-176.
[122] Evins, R., Multi-level optimization of building design, energy system sizing and operation. *Energy* **2015,** 90, 1775-1789.
[123] Khajehpour, S.; Grierson, D. E., Profitability versus safety of high-rise office buildings. *Structural and Multidisciplinary Optimization* **2003,** 25, (4), 279-293.
[124] Rodrigues, E.; Gaspar, A. R.; Gomes, Á., An evolutionary strategy enhanced with a local search technique for the space allocation problem in architecture, Part 1: Methodology. *CAD Computer Aided Design* **2013,** 45, (5), 887-897.
[125] Rodrigues, E.; Gaspar, A. R.; Gomes, Á., An evolutionary strategy enhanced with a local search technique for the space allocation problem in architecture, Part 2: Validation and performance tests. *Computer-Aided Design* **2013,** 45, (5), 898-910.
[126] Song, H.; Ghaboussi, J.; Kwon, T. H., Architectural design of apartment buildings using the Implicit Redundant Representation Genetic Algorithm. *Automation in Construction* **2016,** 72, 166-173.

[127] Yazici, S., A parametric landscape urbanism method: The search for an optimal solution. *A/Z ITU Journal of the Faculty of Architecture* **2016,** 13, (3), 155-165.
[128] Jo, J. H.; Gero, J. S., Space layout planning using an evolutionary approach. *Artificial Intelligence in Engineering* **1998,** 12, (3), 149-162.
[129] Wong, S. S. Y.; Chan, K. C. C., EvoArch: An evolutionary algorithm for architectural layout design. *CAD Computer Aided Design* **2009,** 41, (9), 649-667.
[130] Güleç Özer, D.; Şener, S. M., User accessibility optimization using genetic algorithm: ADA. *A/Z ITU Journal of the Faculty of Architecture* **2013,** 10, (2), 212-230.
[131] Dino, I. G., An evolutionary approach for 3D architectural space layout design exploration. *Automation in Construction* **2016,** 69, 131-150.
[132] Cubukcuoglu, C.; Chatzikonstantinou, I.; Tasgetiren, M. F.; Sariyildiz, I. S.; Pan, Q.-K., A Multi-Objective Harmony Search Algorithm for Sustainable Design of Floating Settlements. *Algorithms* **2016,** 9, (3), 51.
[133] Bahrehmand, A.; Batard, T.; Marques, R.; Evans, A.; Blat, J., Optimizing layout using spatial quality metrics and user preferences. *Graphical Models* **2017,** 93, 25-38.
[134] Li, L., The optimization of architectural shape based on Genetic Algorithm. *Frontiers of Architectural Research* **2012,** 1, (4), 392-399.
[135] Elshaer, A.; Bitsuamlak, G.; El Damatty, A., Enhancing wind performance of tall buildings using corner aerodynamic optimization. *Engineering Structures* **2017,** 136, 133-148.
[136] Deb, K.; Jain, H., An evolutionary many-objective optimization algorithm using reference-point-based nondominated sorting approach, part I: Solving problems with box constraints. *IEEE Trans. Evolutionary Computation* **2014,** 18, (4), 577-601.
[137] Wolpert, D. H.; Macready, W. G., No free lunch theorems for optimization. *IEEE transactions on evolutionary computation* **1997,** 1, (1), 67-82.
[138] Coello, C. A. C., Theoretical and numerical constraint-handling techniques used with evolutionary algorithms: a survey of the state of the art. *Computer methods in applied mechanics and engineering* **2002,** 191, (11-12), 1245-1287.

	INTRODUCTION — METHOD(S)	CHALLENGE(S)
	LITERATURE REVIEW	
J1 **RQ1**	Optimising form-finding parameters in performative computational architecture	■ 17 Form-finding parameters ● 13 Performance aspects ▲ 12 Evolutionary algorithms 3 Swarm algorithms
	TOOL DEVELOPMENT AND PILOT STUDY	
J2 **RQ2**	PART A **Developing Optimus tool using self-adaptive ensemble evolutionary algorithm**	TEST 1 ■ 30 Parameters ▲ 4 Optimisation algorithms
		TEST 2 ■ 70 Parameters ▲ 4 Optimisation algorithms
J3 **RQ3**	PART B **Preliminary results of multi-zone approach using pilot high-rise model**	TEST 3 ■ 100 Parameters ● 2 Daylight metrics ▲ 5 ANN models 1 Optimisation algorithm
	METHODOLOGICAL FRAMEWORK	
J4 **RQ4**	PART A Introducing multi-zone optimisation (MUZO) methodology and prediction results of quad-grid and diagrid high-rise scenarios	TEST 4 ■ 260 Parameters ● 2 Daylight metrics ▲ 20 ANN models
		TEST 5 ■ 220 Parameters ● 2 Daylight metrics ▲ 20 ANN models
J5 **RQ5**	PART B Optimising high-rise scenarios using the predictive models with Optimus and validation of the MUZO methodology	TEST 4 ■ 260 Parameters ● 20 Predictive models ▲ 3 Optimisation algorithms
		TEST 5 ■ 220 Parameters ● 20 Predictive models ▲ 3 Optimisation algorithms
	CASE STUDY	
J6 **RQ6**	Optimising Europoint complex for self-sufficiency in energy consumption and food production using MUZO and Optimus	■ 117 Parameters ● 1 Self-sufficiency in energy 1 Self-sufficiency in food 1 Daylight metric ▲ 45 ANN models 13 Optimisation algorithms
	CONCLUSIONS	

J Journal
RQ Research question

■ Parameter ● Performance ▲ AI Method

3 Optimus tool and pilot high-rise model

Part A of Chapter 3 has been published as: Cubukcuoglu, C.; Ekici, B.; Tasgetiren, M. F.; Sariyildiz, S., OPTIMUS: self-adaptive differential evolution with ensemble of mutation strategies for grasshopper algorithmic modeling. Algorithms 2019, 12, (7), 141. Part B of Chapter 3 has been published as: Ekici, B.; Kazanasmaz, T.; Turrin, M.; Tasgetiren, M. F.; Sariyildiz, I. S. A Methodology for daylight optimisation of high-rise buildings in the dense urban district using overhang length and glazing type variables with surrogate modelling, Journal of Physics: Conference Series, 2019; IOP Publishing: 2019; p 012133. For consistency of the dissertation, the layout is adapted to fit the template, some typos are adjusted, phrases are reworded without changing the content, and the spelling of Part A is converted to British English.

Part A: https://doi.org/10.3390/a12070141
Part B: https://doi.org/10.1088/1742-6596/1343/1/012133

This chapter presents the development of the optimisation tool called Optimus and the pilot study to test the efficiency of the multi-zone optimisation approach in high-rises. Part A presents the Optimus tool, which considers a self-adaptive ensemble evolutionary algorithm that can cope with large numbers of design parameters. Tests 1 and 2 are presented to indicate the relevance of the developed tool based on 30-dimensional Congress on Evolutionary Computation 2005 benchmark problems and a 70-dimensional design problem. Part B explains Test 3 to utilise the efficiency of the multi-zone optimisation approach. The main idea of this method is to divide the building into several subdivisions (zones) to be considered different optimisation problems. The pilot high-rise model considers one of the most used façade parameters reported in Chapter 2 (overhang length) and glazing type for two conflicting daylight metrics predicted by the basic version of artificial neural network models and optimised by the initial version of Optimus tool.

INTRODUCTION

	METHOD(S)	CHALLENGE(S)	
	LITERATURE REVIEW		
J1 RQ1	Optimising form-finding parameters in performative computational architecture	■ 17 Form-finding parameters ● 13 Performance aspects ▲ 12 Evolutionary algorithms 3 Swarm algorithms	
	TOOL DEVELOPMENT AND PILOT STUDY		
J2 RQ2	PART A Developing Optimus tool using self-adaptive ensemble evolutionary algorithm	■ 30 Parameters ▲ 4 Optimisation algorithms	TEST 1
		■ 70 Parameters ▲ 4 Optimisation algorithms	TEST 2
J3 RQ3	PART B Preliminary results of multi-zone approach using pilot high-rise model	■ 100 Parameters ● 2 Daylight metrics ▲ 5 ANN models 1 Optimisation algorithm	TEST 3
	METHODOLOGICAL FRAMEWORK		
J4 RQ4	PART A Introducing multi-zone optimisation (MUZO) methodology and prediction results of quad-grid and diagrid high-rise scenarios	■ 260 Parameters ● 2 Daylight metrics ▲ 20 ANN models	TEST 4
		■ 220 Parameters ● 2 Daylight metrics ▲ 20 ANN models	TEST 5
J5 RQ5	PART B Optimising high-rise scenarios using the predictive models with Optimus and validation of the MUZO methodology	■ 260 Parameters ● 20 Predictive models ▲ 3 Optimisation algorithms	TEST 4
		■ 220 Parameters ● 20 Predictive models ▲ 3 Optimisation algorithms	TEST 5
	CASE STUDY		
J6 RQ6	Optimising Europoint complex for self-sufficiency in energy consumption and food production using MUZO and Optimus	■ 117 Parameters ● 1 Self-sufficiency in energy 1 Self-sufficiency in food 1 Daylight metric ▲ 45 ANN models 13 Optimisation algorithms	

CONCLUSIONS

J Journal
RQ Research question

■ Parameter ● Performance ▲ AI Method

TOC

PART A

OPTIMUS: Self-Adaptive Differential Evolution with Ensemble of Mutation Strategies for Grasshopper Algorithmic Modelling

ABSTRACT

Most of the architectural design problems are basically real-parameter optimisation problems. So, any type of evolutionary and swarm algorithms can be used in this field. However, there is little attention on using optimisation methods within computer-aided design (CAD) programs. In this paper, we present Optimus, which is a new optimisation tool for grasshopper algorithmic modelling in Rhinoceros CAD software. Optimus implements a self-adaptive differential evolution algorithm with an ensemble of mutation strategies (jEDE). We made an experiment using standard test problems in the literature and some of the test problems proposed in IEEE CEC 2005. We reported minimum, maximum, average, standard deviations and number of function evaluations of five replications for each function. Experimental results on the benchmark suite showed that Optimus (jEDE) outperformed other optimisation tools, namely Galapagos (genetic algorithm), SilverEye (particle swarm optimisation), and Opossum (RbfOpt), by finding better results for 19 out of 20 problems. For only one function, Galapagos presented a slightly better result than Optimus. Ultimately, we presented an architectural design problem and compared the tools for testing Optimus in the design domain. We reported minimum, maximum, average and number of function evaluations of one replication for each tool. Galapagos and Silvereye presented infeasible results, whereas Optimus and Opossum found feasible solutions. However, Optimus discovered a much better fitness result than Opossum. In conclusion, we discuss the advantages and limitations of Optimus in comparison to other tools. The target audience of this paper is frequent users of parametric design modelling, e.g., architects, engineers, designers. The main contribution of this paper is summarised as follows. Optimus showed that near-optimal solutions of architectural design problems could be improved by testing different types of algorithms with respect to the no-free lunch theorem. Moreover, Optimus facilitates the implementation of different types of algorithms due to its modular system.

KEYWORDS

Grasshopper, optimisation, differential evolution, architectural design, computational design, performance-based design, building performance optimisation, single-objective optimisation, architectural design optimisation, parametric design

3.1 Introduction: Optimus tool

3.1.1 The necessity of optimisation in architecture

Architectural design problems have an excessive number of design parameters. All possible combinations of design parameters correspond to thousands (in some cases billions) of different design alternatives. Choosing a desirable design alternative within such a big search space is a difficult task. In addition, architectural design requires different performance aspects to be satisfied as design objectives considering the integration of various topics (e.g., social, economic, physiological, health, safety, structural, cultural, sustainability, etc.) [1]. Some of the performance aspects (e.g., energy and daylight) require non-linear equations, which increase the level of complexity. Therefore, decision-making is highly important in the early stages of the design process. Because the decisions that are taken in the early design phases have a great impact on the following design stages. As a result, decision-making in the conceptual phase can be defined as a process that influences the overall performance and the appearance of the constructed building. At this point, computational optimisation techniques have become a necessity in architectural design.

Regarding the computational optimisation techniques, metaheuristic algorithms can play a vital role in not only presenting promising design alternatives but also in dealing with complexity [2]. On the other hand, these algorithms do not guarantee to find globally optimal solutions [3]. However, they can present near-optimal results within a reasonable time. For a decision-maker, providing a near-optimal solution in a reasonable time can be more advantageous than presenting the optimal solution within an extremely long time. Metaheuristics are usually inspired by the natural processes (such as interactions within swarms and evolution over the generations) to discover desirable solutions. Some of these algorithms are harmony search (HS) [4], particle swarm optimisation (PSO) [5], differential evolution (DE) [6,7], genetic algorithm (GA) [8], ant colony optimisation (ACO) [9], simulated annealing (SA) [10], and evolutionary algorithm (EA) [11]. According to the current state of the art, evolutionary computation and swarm optimisation algorithms are the most popular metaheuristics in the architectural domain [2].

3.1.2 Performative computational architecture framework

In order to investigate how the desirable solution can be found in the early phase of the design process, a general framework called performative computational architecture (PCA) [1,2] is considered in this paper. PCA proposes an iterative method based on form-finding, performance evaluation, and optimisation, as illustrated in Fig. 3.1. The form-finding stage includes the geometric generation using parameters in algorithmic modelling environments. The performance evaluation stage comprises the numeric assessments of performance aspects to evaluate how well the form meets the objectives. The optimisation stage corresponds to the metaheuristic algorithms for discovering desirable design solutions within a systematic search process.

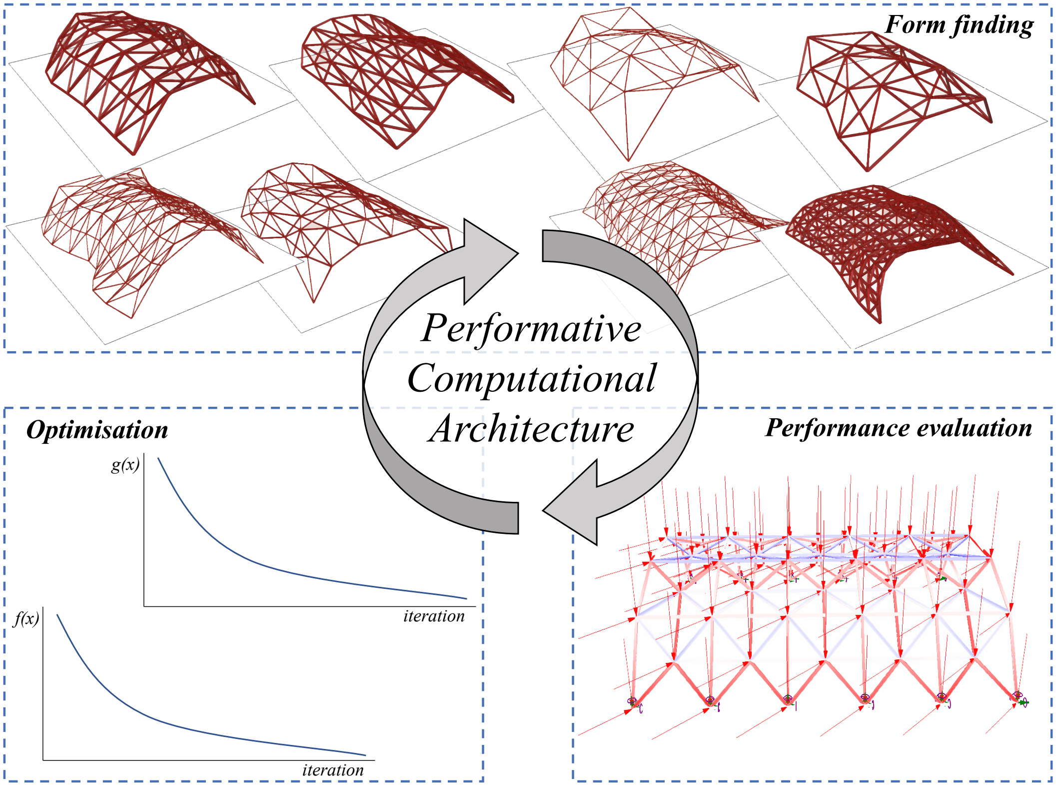


FIG. 3.1 Performative computational architecture (PCA) framework

3.1.3 Current optimisation tools in grasshopper

In this section, existing single-objective optimisation tools for Grasshopper 3d (GH) (available in www.food4rhino.com) are reviewed. Algorithm applications for specific optimisation problems (such as topology optimisation for structure) are not considered. In this context, Galapagos, Goat, Silvereye, Opossum, Dodo, and Nelder–Mead optimisation plug-ins, shown in Fig. 3.2, are explained. Some of these plug-ins have been compared in building optimisation problems in the literature, which can be found in [12–15].

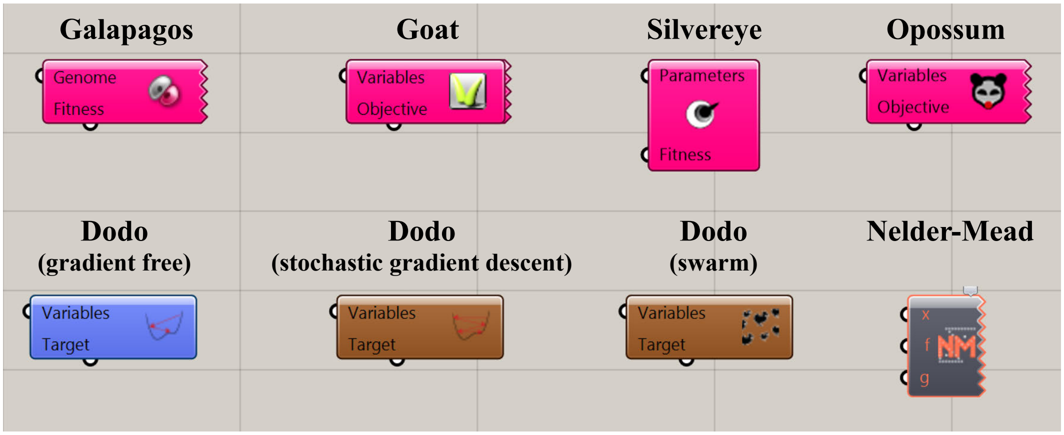


FIG. 3.2 Existing single-objective optimisation plug-ins in grasshopper (GH)

3.1.3.1 Galapagos

Galapagos [16] is one of the first released optimisation plug-ins for GH. The tool provides two heuristic optimisation algorithms, which are GA [8] and SA [10]. The developer of the tool suggests SA for rough landscape navigation, whereas evolutionary solver for finding reliable intermediate solutions. The majority of the design optimisation papers in the literature utilised the Galapagos tool in dealing with energy [17–19], daylight [20,21], both energy and daylight [22,23] and structure [24–26].

TOC

3.1.3.2 Goat

Goat [27] uses the NLopt library [28] in the graphical user interface of Galapagos. The tool considers a mathematically-rigorous approach (gradient-free optimisation algorithm) to reach fast and deterministic results. The Goat provides several optimisation algorithms that are constrained optimisation by linear approximation (COBYLA), bound optimisation by quadratic approximation (BOBYQA), subplex algorithm (Sbplx), the dividing rectangles algorithm (DIRECT), and controlled random search 2 (CRS2). Very recently, Goat has been used for building energy optimisation [14] and structure optimisation [29,30].

3.1.3.3 Silvereye

Despite the gradient-free optimisation and evolutionary computation, Silvereye [15] is one of the swarm intelligence optimisation plug-ins released for GH. The tool considers ParticleSwarmOptimization.dll, which is a shared library containing an implementation of the core version of the PSO. In the literature, Silvereye is used in the design optimisation problems that focus on energy [14], microclimate [31] and structural [29].

3.1.3.4 Opossum

Opossum [32] is the first model-based optimisation tool for GH. The solver is based on an open-source library for black-box optimisation with costly function evaluations (RBFOpt) [33], such as energy and daylight simulations. RBFOpt library uses the radial basis function with the local search while discovering satisfactory solutions with a small number of function evaluations. The Opossum is used in several design problems to deal with daylight [34], structure [29,30] and energy [14].

3.1.3.5 Dodo

Dodo [35] is another plug-in based on the various implementations of optimisation algorithms that are non-linear gradient-free optimisation based on NLopt library [28], stochastic gradient descent algorithm, and swarm optimisation. In addition, Dodo also provides several supervised and unsupervised neural network algorithms.

3.1.3.6 Nelder-Mead optimisation

Nelder–Mead Optimisation [36] is the first tool based on the Nelder–Mead method [37], a local search-based optimisation algorithm in GH. Compared to heuristics, Nelder–Mead typically has fewer function evaluations for computationally expensive models. In addition, the implementation of Nelder–Mead Optimisation also allows considering multiple constraints using the Kreisselmeier-Steinhauser function [38].

3.1.4 This study: Optimus

In the field of computer science, different types of metaheuristic algorithms have been suggested to solve real-parameter optimisation problems by researchers and engineers for many years. As a common approach, the performance of each developed algorithm is tested by using a set of the standard benchmark problems such as Sphere, Schwefel's, Rosenbrock's, Rastrigin's, etc. For real-world problems, this test is done by considering benchmark instances. Moreover, performances of proposed algorithms are frequently compared with existing algorithms because of the no free lunch (NFL) theorem [39]. According to the NFL theorem, the performance of an optimisation algorithm depends on the nature of the problem. In other words, one algorithm can outperform another algorithm in a specific problem. Thus, the NFL theorem argues that there is no global optimisation algorithm that can present the best result for all real-world and benchmark problems.

In the field of architecture, testing different types of algorithms for the same architectural design problem is not a common approach. One of the most important reasons for this fact is that computer-aided design (CAD) tools of architects do not include optimisation algorithms in a wide range. According to the current state of the art [40], only 3% of total users of optimisation tools are architects in the domain of building performance optimisation. Therefore, one may argue that there is little attention on using optimisation methods within the CAD programs. This paper introduces a new optimisation solver, called Optimus, with significant features listed below:

- Compatible with parametric design models created in GH [16] algorithmic modelling for Rhinoceros [41] CAD software.
- Supports PCA framework as outlined in the previous section.
- Implements a self-adaptive [42] differential evolution algorithm with an ensemble of mutation strategies [43] (jEDE), explained in Section 3.3.
- Achieves the highest performance when compared to other optimisation algorithms available in GH reviewed in Section 3.1.3.

The performance of the Optimus is tested by benchmark suite, which consists of standard single-objective unconstrained problems and some of the test problems proposed in IEEE Congress on Evolutionary Computation 2005 (CEC 2005) [44], and a design problem having large numbers of parameters. The proposed algorithm is explained in Section 3.2, the implementation for GH is presented in Section 3.3, and experimental results are given in Section 3.4. Problem formulations and optimisation results of the benchmarks are given in Section 3.4.1, whereas Section 3.4.2. presents the formulation and optimisation results of the design problem. The development process of the Optimus is illustrated in Fig. 3.3.

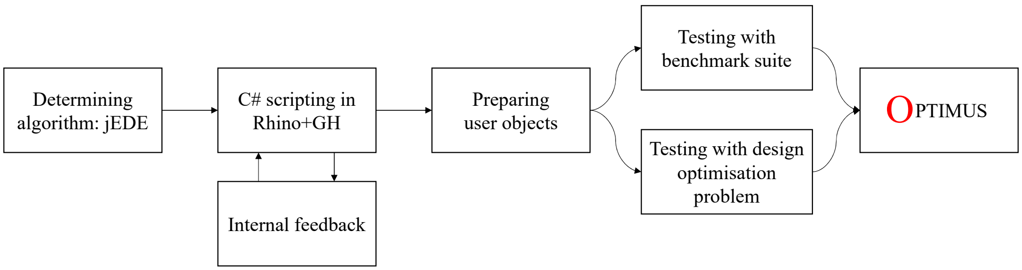


FIG. 3.3 Optimus development process

3.2 Self-adaptive differential evolution algorithm with ensemble of mutation strategies

Metaheuristics are one of the most used optimisation methods in the domain of architectural design [2]. These algorithms can avoid local minima and maxima while coping with real parameters in large search spaces. In addition, metaheuristics can present near-optimal results when compared to other direct search methods in a reasonable time [3].

Swarm intelligence (SI) and evolutionary algorithms (EAs) are the most common sub-categories of metaheuristics that are inspired by nature using different search strategies. SI is based on interactions of swarms such as flocks of birds, schools of fish, ants and bees. Some examples of SI can be shown as Ant Colony Optimisation

TOC

(ACO) [9] and Particle Swarm Optimisation (PSO) [45,46]. On the other hand, EAs are in the class of population-based metaheuristic optimisation algorithms that are inspired by the mechanisms of biological evolution that mimic the selection and reproduction processes of living organisms. EAs are very effective to deal with NP-hard problems. Some examples of EAs are Genetic Algorithms (GAs) [8,47]. Genetic Programming (GP) [48], Evolution Strategy (ES) [49], and DE [7].

In the EA category, DE, which is introduced by Storn and Price [7], is potentially one of the most powerful stochastic real-parameter optimisation algorithms in the literature. The algorithm can converge very fast in solving real-world problems such as in the domain of scheduling [50], optics [51], communication [6], power systems [52], pattern recognition [53] and recently in architectural design [54,55]which is a rural touristic region located on the west coast of Turkey, near the metropolis of Izmir. The problem at hand includes both engineering and architectural aspects that need to be addressed in a comprehensive manner. We thus adapt the view as a multi-objective constrained real-parameter optimization problem. Specifically, we consider three objectives, which are conflicting. The first one aims at maximizing accessibility of urban functions such as housing and public spaces, as well as special functions, such as a marina for yachts and a yacht club. The second one aims at ensuring the wind protection of the general areas of the settlement, by adequately placing them in between neighboring land masses. The third one aims at maximizing visibility of the settlement from external observation points, so as to maximize the exposure of the settlement. To address this complex multi-objective optimization problem and identify lucrative alternative design solutions, a multi-objective harmony search algorithm (MOHS).

A recent survey by Das and Suganthan [56,57] clearly explains the history of DE and its success. One of the advantages of the DE algorithm is the simple code structure that facilitates its implementation. Another advantage is that the number of control parameters in DE is few, which are crossover rate (CR), mutation rate (MR), and population size (NP), when compared to other metaheuristic algorithms. In classical DE, these control parameters are constant during the whole optimisation process. However, a simple change in MR or CR generation strategies can significantly improve the performance of the algorithm. Therefore, some variants of DE in the literature focus on parameter settings as presented in [42,58]. Moreover, DE can tackle the large scale and computationally expensive optimisation problems as the space complexity of DE is low as mentioned in [59].

3.2.1 The basic differential evolution

The classical DE has four main stages. There are recursive processes among the second, third and fourth steps as follows:

- Initialisation for generating the initial target population once at the beginning.
- Reproduction with mutation for generating the mutant population by using the target population.
- Reproduction with crossover for generating the trial population by using the mutant population.
- Selection to choose the next generation among trial and target populations using one-to-one comparison. In each generation, individuals of the current population become the target population.

3.2.1.1 Initialisation

In the basic DE algorithm, the initial target population has NP individuals with a D-dimensional real-parameter vectors. Each vector is obtained randomly and uniformly within the search space restricted by the given minimum and maximum bounds: $\left[x_{ij}^{\min}, x_{ij}^{\max}\right]$. Thus, the initialisation of j^{th} component of i^{th} vector can be defined as:

$$x_{ij}^{0} = x_{ij}^{\min} + \left(x_{ij}^{\max} - x_{ij}^{\min}\right) \times r \,, \tag{3.1}$$

where x_{ij}^{0} is the i^{th} target population at generation $g = 0$; and r is a uniform random number in the range $[0,1]$.

3.2.1.2 Mutation

The difference vector in the mutation operator is one of the main strengths of DEs [7]. Hence, DE differs from other EAs since it relies on a difference vector with a scale factor MR. The mutation process is the first step to generate new solutions. In order to obtain mutant population, two individuals are randomly chosen from the target population.

The weighted difference of these individuals is added to a third individual from the target population as in Equation (3.2).

$$v_{ij}^{g} = x_{kj}^{g-1} + MR \times \left(x_{lj}^{g-1} - x_{mj}^{g-1} \right), \tag{3.2}$$

where k, l, m are three randomly chosen individuals from the target population such that $\left(k \neq l \neq m \neq i \in (1, \ldots, NP)\right)$ and $j = 1, \ldots, D$. $MR > 0$ is a mutation scale factor influencing the differential variation between two individuals. v_{ij}^{g} is the mutant population in generation g.

3.2.1.3 Crossover

To obtain the trial population, a binomial crossover is applied to each variable. If a randomly and uniformly generated number $r\,[0,1]$ is less than or equal to the crossover rate (CR), the individuals from the mutant population are chosen; otherwise, target individuals are selected. Simply, the trial population is generated by recombining mutant individuals with their corresponding target individuals as follows:

$$u_{ij}^{g} = \begin{cases} v_{ij}^{g} & if \;\; r_{ij}^{g} \leq CR \;\; or \;\; j = D_j \\ x_{ij}^{g-1} & otherwise \end{cases}, \tag{3.3}$$

where the index D_j is a randomly chosen dimension from $(j = 1, \ldots, D)$. It makes sure that at least one parameter of the trial population u_{ij}^{g} will be different from the target population x_{ij}^{g-1}. CR is a user-defined crossover constant in the range $[0,1]$, and r_{ij}^{g} is a uniform random number in the interval $[0,1]$ whereas u_{ij}^{g} is the trial population at generation g.

When the trial population is obtained, parameter values might violate search boundaries. Therefore, the solution can be restricted. For this reason, parameter values that violate the search range are randomly and uniformly re-generated as in Equation (3.4).

$$x_{ij}^{g} = x_{ij}^{\min} + \left(x_{ij}^{\max} - x_{ij}^{\min}\right) \times r \ . \tag{3.4}$$

3.2.1.4 Selection

For the next generation, the selection process is realised, which is based on the survival of the fittest among the trial and target populations. The population that has the lower fitness value is chosen according to one-to-one comparison, as in Equation (3.5).

$$x_{i}^{g} = \begin{cases} u_{i}^{g} & if \;\; f\left(u_{i}^{g}\right) \leq f\left(x_{i}^{g-1}\right) \\ x_{i}^{g-1} & otherwise \end{cases} . \tag{3.5}$$

3.2.2 Self-adaptive approach

In this paper, self-adaptive DE [42], so-called *jDE*, is employed. The jDE is very simple, effective and converges much faster than the basic DE, especially when the dimensionality of the problem is high or the problem is complex. However, in the jDE, each individual has their own MR_i and CR_i values. In this paper, these values are initially taken as $CR_i = 0.5$ and $MR_i = 0.9$ and they are updated as follows:

$$MR_{i}^{g} = \begin{cases} MR_l + r_1 \times MR_u & if \;\; r_2 < t_1 \\ MR_{i}^{g-1} & otherwise \end{cases} , \tag{3.6}$$

$$CR_{i}^{g} = \begin{cases} r_3 & if \;\; r_4 < t_2 \\ CR_{i}^{g-1} & otherwise \end{cases} , \tag{3.7}$$

where $r_j \in \{1,2,3,4\}$ are uniform random numbers in the range $[0,1]$. t_1 and t_2 represent the probabilities to adjust the MR and CR. They are taken as $t_1 = t_2 = 0.1$ and $MR_l = 0.1$ and $MR_u = 0.9$.

3.2.3 Ensemble approach

In addition to the self-adaptive approach, an ensemble approach [43] is employed in the jDE, so-called jEDE. This means that instead of using one type of mutation strategy with a fixed parameter setting as in the basic DE, each mutant individual is generated according to different mutation strategies with different parameter settings. In this approach, each dimension has a value pool for a competition of producing better future offspring according to their success in the past generations. In this study, the following mutation strategies (M_i) are considered as in Equations (3.8)–(3.10). In M_1, the individuals that formed the mutant population are randomly selected. In M_2 and M_3, strategies are benefitted from the information of the best solution (xbest) so far.

$$if \;\; STR_i = 0 \quad M_1 : v_i^{j,t+1} = x_k^{j,t} + MR_i \times \left(x_l^{j,t} - x_m^{j,t} \right), \tag{3.8}$$

$$if \;\; STR_i = 1 \quad M_2 : v_i^{j,t+1} = x_{best}^{j,t} + MR_i \times \left(x_l^{j,t} - x_m^{j,t} \right), \tag{3.9}$$

$$if \;\; STR_i = 2 \quad M_3 : v_i^{j,t+1} = x_i^{j,t} + MR_i \times \left(x_{best}^{j,t} - x_i^{j,t} \right) + F \times \left(x_k^{j,t} - x_l^{j,t} \right), \tag{3.10}$$

where k, l, m are three randomly selected individuals from the target population such that $\left(k \neq l \neq m \neq i \in (1, \ldots, NP) \right)$ and $j = 1, \ldots, D$. $MR_i > 0$ is a mutation scale factor; in jEDE of this study, it is generated by using a self-adaptive procedure. STR_i is the strategy used in each population to choose different mutation strategies. The pseudo-code of the final jEDE algorithm is given in Algorithm 1.

Algorithm 1. The self-adaptive differential evolution algorithm with ensemble of mutation strategies.

Set parameters $g = 0$, $NP = 100$, $M_{\max} = 4$

Establish initial population randomly

$P^g = \left\{ x_1^g, \ldots, x_{NP}^g \right\} \;\; with \;\; x_i^g = \left\{ x_{i1}^g, \ldots, x_{iD}^g \right\}$

Assign a mutation strategy to each individual randomly

$M_i = rand(\;)\% M_{\max} \;\; for \;\; i = 1, \ldots, NP$

6 Evaluate population and find x_{best}^{g}

7 $f\left(P^{g}\right)=\left\{f\left(x_{1}^{g}\right),\ldots,f\left(x_{NP}^{g}\right)\right\}$

8 Assign $CR[i]=0.5$ and $CR[i]=0.9$ to each individual

9 Repeat the following for each individual x_{i}^{g}

10 Obtain $v_{i}^{g}=M_{i}\left(x_{i}^{g}\right)$

11 Obtain $u_{i}^{g}=CR_{i}\left(x_{i}^{g},v_{i}^{g}\right)$

12 Obtain $x_{i}^{g}=\begin{cases} u_{i}^{g} & if\ \ f\left(u_{i}^{g}\right)\leq f\left(x_{i}^{g-1}\right) \\ x_{i}^{g-1} & otherwise \end{cases}$

13 $If\ \ f\left(u_{i}^{g}\right)>f\left(x_{i}^{g-1}\right),\quad M_{i}=rand(\ )\%M_{\max}$

14 $If\ \ f\left(x_{i}^{g}\right)\leq f\left(x_{best}^{g}\right),\quad x_{best}^{g}=x_{i}^{g}$

15 Update F_{i}^{g} and CR_{i}^{g}

16 If the stopping criterion is not met, go to Lines 9–15

17 Else stop and return π_{best}

3.3 Optimus

Optimus is a new optimisation plug-in (https://www.food4rhino.com/app/optimus) developed for GH. The beta version (1.0.0) of the plug-in implements the self-adaptive [42] differential evolution algorithm with an ensemble of mutation strategies [43] (jEDE) explained in the previous section. The algorithm is coded in C#, which is one of the available programming languages for custom scripting components in the GH. Optimus is based on a modular approach with many C# items. Every step of the optimisation process can be observed within these items. Fig. 3.4 shows the eight components of Optimus, which are categorised under three main titles, as follows:

1 Optimus consists of

a ReadMe (giving details about Optimus),

b jEDE (using jEDE algorithm, generating mutant and trial populations)

c One-to-one (enabling selection process for next generation)

2 Initialise consists of

a GetBound (taking the boundaries of design variables in D dimensions)

b InitPop (generating initial population for NP population size in D dimensions)

3 Utilities consist of

a xBest (finding chromosomes that have the lowest fitness value in the population)

b Merge (collecting the fitness, population, xBest and optimisation parameters)

c Unmerge (separating the fitness, population, xBest and optimisation parameters)

To manage the main loop of the Optimus, the HoopSnake component [60] is used for enabling feedback loops (recursive executions) within the GH.

In the Optimus, each of the eight components is transferred to GH clusters by defining collections of inputs and outputs, as shown in Fig. 3.5. This gives flexibility to the user for improving the C# code according to the user's implementation purposes. As the next step, each cluster is converted to GH user object files, which are provided in the supplementary materials to be used as a plug-in for the GH.

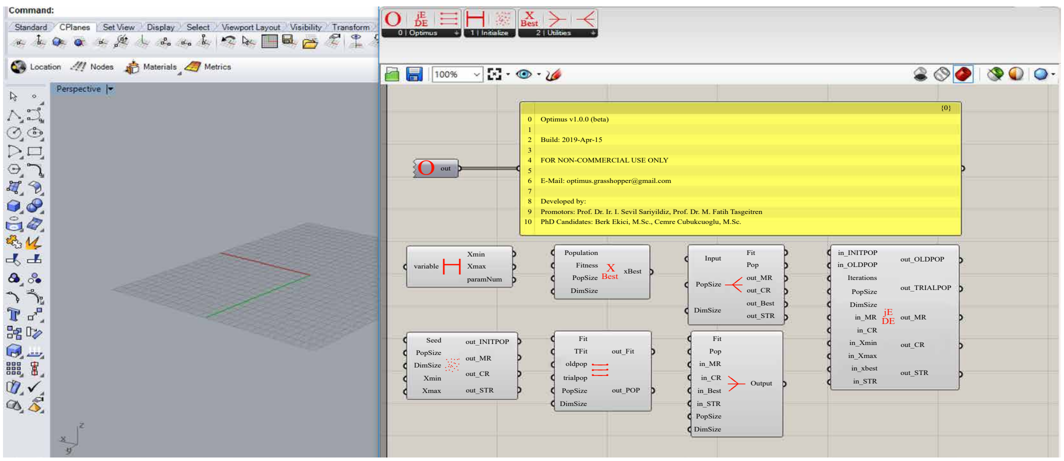


FIG. 3.4 Components of Optimus v1.0.0 (beta)

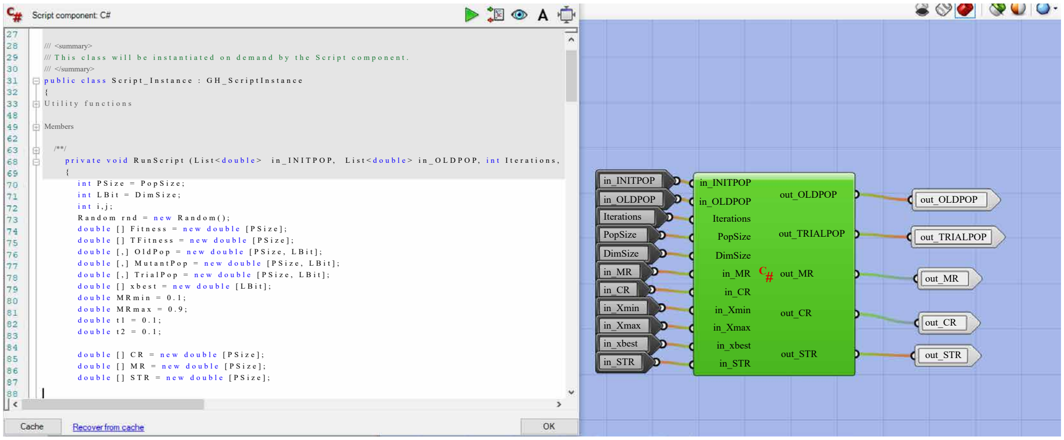


FIG. 3.5 Example cluster implementation for self-adaptive differential evolution algorithm with ensemble of mutation strategies (jEDE) component

The user needs to follow several steps for using Optimus:

1. Place GetBound on the GH canvas and connect with number sliders.
2. Define the population size.
3. Get InitPop for initialisation using population size and output of GetBound.
4. Evaluate initial fitness using the output of InitPop.
5. Internalise the initial fitness.
6. Place xBest on the GH canvas.
7. Get Merge and connect with internalised initial fitness and outputs of InitPop and xBest.
8. Connect Merge with starting input (S) of HoopSnake.
9. Place UnMerge on the GH canvas and connect with feedback output (F) of HoopSnake.
10. Get jEDE and connect outputs of UnMerge, InitPop, GetBound.
11. Evaluate trial fitness using the output of jEDE.
12. Get One-to-One and connect with initial fitness, trial fitness and outputs of jEDE.
13. Place xBest and connect outputs of One-to-one for updating the new best chromosome.
14. Get Merge and connect outputs of jEDE, One-to-one, and xBest.
15. Complete the loop by connecting the output of Merge to the data input (D) of the HoopSnake.
16. These steps are visualised in Fig. 3.6, and an example file is shared in supplementary materials.

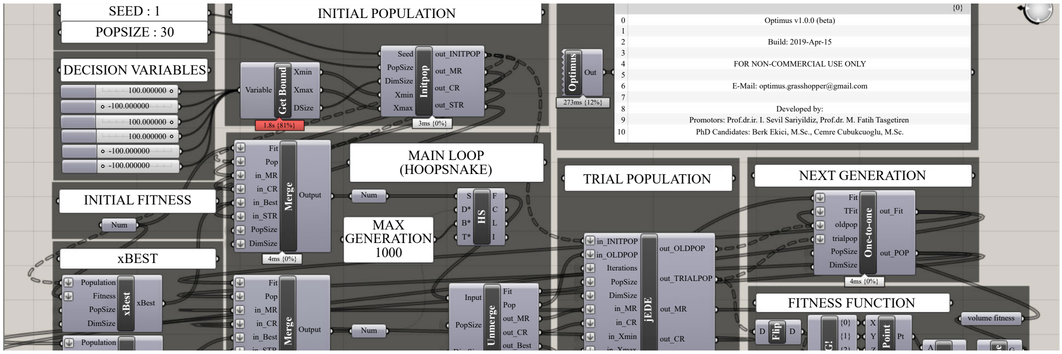


FIG. 3.6 Visualisation of the Optimus loop

To save some computation time, Optimus does not update number sliders in the GH. During the optimisation process of the design problem, the geometry is generated with initial and trial populations. For this reason, NP size of geometries is observed in each iteration. During the initial stage, various types of design alternatives are generated. However, when the Optimus is converged, similar geometries are observed, as shown in Fig. 3.7.

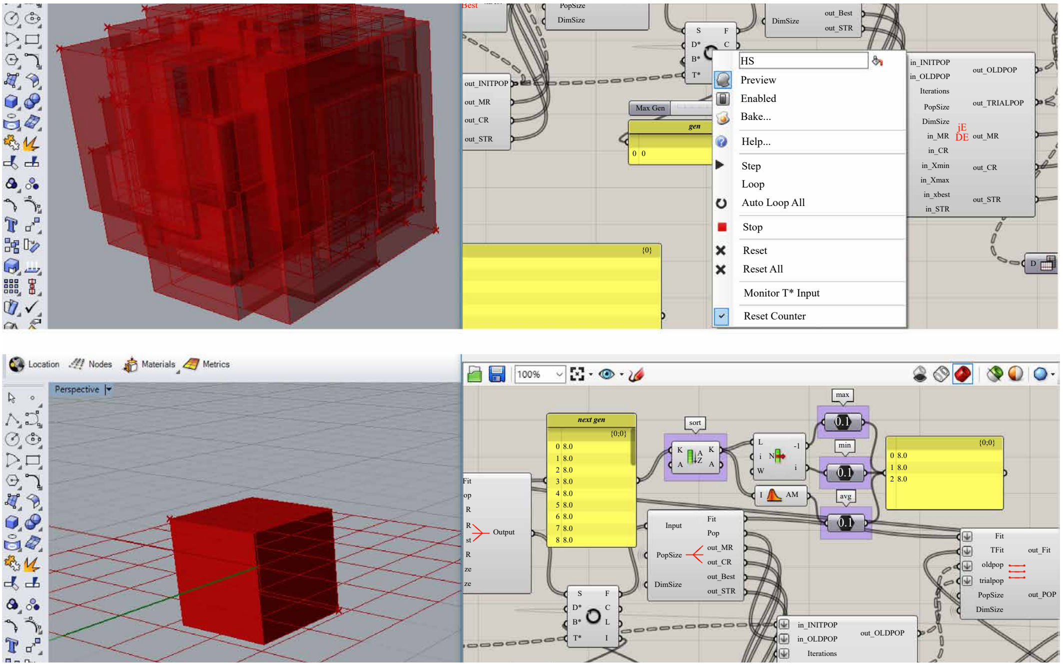


FIG. 3.7 Examples of initial and optimised populations

TOC

3.4 Experiments

This section explains how the experiments were performed to test the performance of the Optimus in comparison to the other chosen plug-ins. As mentioned before, the evaluations were completed by using standard benchmark problems in the literature and one architectural design problem proposed by the authors of this paper.

3.4.1 Benchmark suite

The performance of the Optimus (jEDE algorithm) was firstly tested by using the following benchmark suite, which consists of 20 test functions. The first ten functions in the benchmark suite were classical test problems that have been commonly used in the literature. The remaining ten functions were taken from the benchmark suite presented in the CEC 2005 Special Session on Real-Parameter Optimisation that has been modified from the classical test problems in order to locate their global optimum under some circumstances (e.g., shifted and/or rotated landscape, optimum placed on bounds, Gaussian noise and/or bias added etc.) [44]. This modification makes these functions more difficult to solve than the classical test functions. In the test suite of this study, F_1 to F_5 are unimodal, F_6 to F_{10} are multimodal functions, whereas all the benchmark functions are minimisation problems. The benchmark suite is presented in Table 3.1, while these functions are summarised in Appendix 3A.

3.4.1.1 Experimental setup and evaluation criteria

All the benchmark functions were coded in C# as individual components in the GH environment. These components are also available in supplementary materials as GH files to contribute evaluations of further developed architectural design optimisation tools. Furthermore, all the benchmark functions ran on a computer that has Intel Core I7-6500U CPU @ 2.50 GHz with 16 GB RAM. Both the number of dimension D and the population size NP were taken as 30 to limit the search space. For a fair comparison, termination criteria were defined as 30 min for each benchmark component. Additionally, five replications were carried out for each function and for each tool (thus, the total run time is 12,000 min).

TABLE 3.1 Benchmark suite

Notation	Function
F_{sph}	Sphere Function
F_{ros}	Rosenbrock's Function
F_{ack}	Ackley's Function
F_{grw}	Griewank's Function
F_{ras}	Rastrigin's Function
F_{sch}	Generalised Schwefel's Problem 2.26
F_{sal}	Salomon's Function
F_{wht}	Whitely's Function
F_{pn1}	Generalised Penalised Function 1
F_{pn2}	Generalised Penalised Function 2
F_1	Shifted Sphere Function
F_2	Shifted Schwefel's Problem 1.2
F_3	Shifted Rotated High Conditioned Elliptic Function
F_4	Shifted Schwefel's Problem 1.2 with Noise in Fitness
F_5	Schwefel's Problem 2.6 With Global Optimum on Bounds
F_6	Shifted Rosenbrock's Function
F_7	Shifted Rotated Griewank's Function without Bounds
F_8	Shifted Rotated Ackley's Function with Global Optimum on Bounds
F_9	Shifted Rastrigin's Function
F_{10}	Shifted Rotated Rastrigin's Function

For evaluating the performance of the algorithms (of the components), we basically reported *min f(x)*, *max f(x)*, *avg f(x)*, and *std f(x)* that are the minimum, maximum, average, and standard deviation values of the function x in completed replications. The maximum number of fitness evaluations (FES) within 30 min for each tool were also recorded, which means how many times the fitness was tested during each 30-min optimisation process.

3.4.1.2 Experimental results

As mentioned before, Galapagos employs GA, SilverEye (v1.1.0) uses PSO and Opossum (v1.5.0) considers RBFOpt for enabling architectural design optimisation in the GH environment. We compared the results of Optimus, which uses jEDE, with those three optimisation tools to present its performance. In addition, all the runs for each component and for each function were taken by the authors. Table 3.2 shows the fitness results of Optimus, Opossum, SilverEye, and Galapagos, together with optimal fitness values of each function. Results clearly indicated the superiority of the Optimus over other optimisation tools such that it yielded the lowest *min f(x)* in nineteen functions out of twenty significantly. On the other hand, Galapagos outperformed Optimus in only one function (F_8) with a very small fitness difference. *max f(x)* and *avg f(x)* values in Table 3.2 further justify the better performance of the Optimus in such a way that the maximum and average function values of Optimus were closer to optimal fitness values. Furthermore, the FES within thirty minutes in each problem were the highest in Optimus. This clearly showed high margins between Optimus and other components, where Optimus was tremendously faster in each iteration. For example, when solving F_{sph} , the Optimus (jEDE) approximately realised 5844 FES/minute, whereas GA made 1643 FES/minute, PSO completed 1205 FES/minute, RBFOpt made 85 FES/minute. These results explicitly implied the superiority of Optimus over other components in solving the benchmark suite.

TABLE 3.2 Comparison of Optimus, Opossum, Silvereye, Galapagos (D = 30, NP = 30, termination: 30 min)

		Optimus_jEDE	Opossum_RBFOpt	SilverEye_PSO	Galapagos_GA	Optimal
F_{sph}	*min f(x)*	0.0000000×10^{0}	1.4000000×10^{-5}	5.9000000×10^{-5}	1.1709730×10^{0}	0
	max f(x)	0.0000000×10^{0}	5.8000000×10^{-5}	2.7057298×10^{1}	4.4052130×10^{0}	
	avg f(x)	0.0000000×10^{0}	3.6400000×10^{-5}	5.4171618×10^{0}	2.7928586×10^{0}	
	std f(x)	0.0000000×10^{0}	1.8039956×10^{-5}	1.0820072×10^{-1}	1.1492298×10^{0}	
	FES	194,520	3225	31,560	34,260	
F_{ros}	*min f(x)*	0.0000000×10^{0}	2.7485056×10^{1}	1.6689612×10^{1}	6.0863438×10^{3}	0
	max f(x)	3.9866240×10^{0}	2.1030328×10^{2}	5.8965910×10^{4}	2.2859534×10^{4}	
	avg f(x)	2.3919744×10^{0}	9.0892522×10^{1}	1.3859753×10^{4}	1.3060872×10^{4}	
	std f(x)	1.9530389×10^{0}	7.1919037×10^{1}	2.2886020×10^{4}	6.7095472×10^{4}	
	FES	149,460	882	26,700	35,070	
F_{ack}	*min f(x)*	0.0000000×10^{0}	3.3550000×10^{-3}	1.3404210×10^{0}	5.7470000×10^{-2}	0
	max f(x)	1.3404210×10^{0}	2.4098540×10^{0}	3.7340120×10^{0}	1.0270860×10^{0}	
	avg f(x)	2.6808420×10^{-1}	1.3795174×10^{0}	2.2482728×10^{0}	4.8037520×10^{-1}	
	std f(x)	5.3616840×10^{-1}	8.5713298×10^{-1}	9.1850828×10^{-1}	4.0392221×10^{-1}	
	FES	206,370	1447	38,490	28,710	
F_{grw}	*min f(x)*	0.0000000×10^{0}	1.5840000×10^{-3}	3.2081000×10^{-2}	3.4407200×10^{-1}	0
	max f(x)	0.0000000×10^{0}	1.7086000×10^{-2}	2.6292800×10^{-1}	1.0657060×10^{0}	
	avg f(x)	0.0000000×10^{0}	7.6638000×10^{-3}	1.2049020×10^{-1}	8.2474220×10^{-1}	
	std f(x)	0.0000000×10^{0}	5.6121253×10^{-3}	8.1064770×10^{-2}	2.6131521×10^{-1}	
	FES	151,110	1089	26,610	37,410	
F_{ras}	*min f(x)*	4.9747950×10^{0}	2.5870757×10^{1}	3.3829188×10^{1}	7.0535550×10^{0}	0
	max f(x)	2.3879007×10^{1}	4.1789542×10^{1}	6.1687356×10^{1}	2.9072445×10^{1}	
	avg f(x)	1.3332448×10^{1}	3.6218407×10^{1}	4.8355074×10^{1}	1.5404780×10^{1}	
	std f(x)	6.7363920×10^{0}	5.4349940×10^{0}	1.1424086×10^{1}	9.1077975×10^{0}	
	FES	206,520	4149	37,650	51,480	
F_{sch}	*min f(x)*	2.3687705×10^{2}	1.2877414×10^{3}	4.1089621×10^{3}	1.9550066×10^{3}	0
	max f(x)	4.7375372×10^{2}	4.0111803×10^{3}	6.1589437×10^{3}	2.7977670×10^{3}	
	avg f(x)	4.0269072×10^{2}	2.7169368×10^{3}	5.2658793×10^{3}	2.3201102×10^{3}	
	std f(x)	9.4750668×10^{1}	8.8809862×10^{2}	6.6677783×10^{2}	2.8681876×10^{2}	
	FES	148,140	1487	27,210	35,940	
F_{sal}	*min f(x)*	1.9987300×10^{-1}	2.8070190×10^{0}	2.9987300×10^{-1}	1.4998750×10^{0}	0
	max f(x)	4.9987300×10^{-1}	4.3000810×10^{0}	4.9987300×10^{-1}	2.8375760×10^{0}	
	avg f(x)	3.1987300×10^{-1}	3.4413810×10^{0}	3.7987300×10^{-1}	2.0682340×10^{0}	
	std f(x)	1.1661904×10^{-1}	5.6623101×10^{-1}	7.4833148×10^{-2}	6.1557512×10^{-1}	
	FES	201,720	4769	38,640	51,360	

>>>

TABLE 3.2 Comparison of Optimus, Opossum, Silvereye, Galapagos (D = 30, NP = 30, termination: 30 min)

		Optimus_jEDE	Opossum_RBFOpt	SilverEye_PSO	Galapagos_GA	Optimal
F_{wht}	*min f(x)*	2.3704633×10^{1}	9.6592754×10^{2}	1.8455490×10^{2}	2.3632742×10^{5}	0
	max f(x)	2.4040716×10^{2}	1.6904059×10^{3}	6.2776811×10^{2}	5.5055000×10^{8}	
	avg f(x)	1.0789137×10^{2}	1.2610498×10^{3}	4.0698894×10^{2}	2.7440867×10^{8}	
	std f(x)	7.4951993×10^{1}	2.6984398×10^{2}	1.6923390×10^{2}	2.2575944×10^{8}	
	FES	146,640	728	23,250	29,730	
F_{pn1}	*min f(x)*	0.0000000×10^{0}	2.9057000×10^{-2}	3.1283800×10^{-1}	1.4510000×10^{-3}	0
	max f(x)	0.0000000×10^{0}	9.0392970×10^{0}	1.3487570×10^{0}	1.7632000×10^{-2}	
	avg f(x)	0.0000000×10^{0}	2.8243854×10^{0}	6.7680180×10^{-1}	6.0638000×10^{-3}	
	std f(x)	0.0000000×10^{0}	3.1774566×10^{0}	4.2868737×10^{-1}	6.0403379×10^{-3}	
	FES	203,880	1394	39,420	57,720	
F_{pn2}	*min f(x)*	0.0000000×10^{0}	2.0400434×10^{1}	$1.0000000 \times 10^{-11}$	1.8037300×10^{-1}	0
	max f(x)	1.0987000×10^{-2}	2.8693232×10^{1}	9.3079800×10^{-1}	2.7208440×10^{0}	
	avg f(x)	2.1974000×10^{-3}	2.5384324×10^{1}	2.2552480×10^{-1}	1.0041520×10^{0}	
	std f(x)	4.3948000×10^{-3}	3.4851206×10^{0}	3.5679494×10^{-1}	9.4298611×10^{-1}	
	FES	148,380	639	29,040	41,520	
F_1	*min f(x)*	-4.5000000×10^{2}	-4.4999898×10^{2}	2.6232595×10^{2}	-4.4995998×10^{2}	-450
	max f(x)	-4.5000000×10^{2}	-4.4999478×10^{2}	8.0377273×10^{3}	-4.4988406×10^{2}	
	avg f(x)	-4.5000000×10^{2}	-4.4999680×10^{2}	4.0824562×10^{3}	-4.4992015×10^{2}	
	std f(x)	0.0000000×10^{0}	1.4829922×10^{-3}	2.9460423×10^{3}	3.0428108×10^{-2}	
	FES	198,060	6156	45,580	66,180	
F_2	*min f(x)*	-4.5000000×10^{2}	3.4590652×10^{4}	-3.8035693×10^{2}	6.8476195×10^{3}	-450
	max f(x)	-4.5000000×10^{2}	4.7978174×10^{4}	5.2590674×10^{2}	1.2302281×10^{4}	
	avg f(x)	-4.5000000×10^{2}	4.3226072×10^{4}	-1.2838464×10^{2}	1.0174618×10^{4}	
	std f(x)	0.0000000×10^{0}	4.9030645×10^{3}	3.3183646×10^{2}	1.8557926×10^{3}	
	FES	146,010	1061	33,840	50,160	
F_3	*min f(x)*	6.0045376×10^{4}	1.5561000×10^{7}	1.9264000×10^{6}	1.1250000×10^{7}	-450
	max f(x)	2.4850013×10^{5}	7.5084000×10^{7}	8.0820000×10^{6}	2.7772000×10^{7}	
	avg f(x)	1.2857393×10^{5}	3.7380600×10^{7}	4.5525000×10^{6}	1.7212200×10^{7}	
	std f(x)	6.4175884×10^{4}	2.0647812×10^{7}	2.0206526×10^{6}	5.6281541×10^{6}	
	FES	205,260	1293	48,030	66,000	
F_4	*min f(x)*	-4.5000000×10^{2}	2.8373782×10^{4}	-4.2715712×10^{2}	7.8877685×10^{3}	-450
	max f(x)	-4.5000000×10^{2}	3.9404224×10^{4}	4.0484178×10^{3}	1.1191542×10^{4}	
	avg f(x)	-4.5000000×10^{2}	3.2359668×10^{4}	5.1724092×10^{2}	9.4535270×10^{3}	
	std f(x)	0.0000000×10^{0}	4.0412734×10^{3}	1.7663033×10^{3}	1.2966977×10^{3}	
	FES	147,240	1055	35,610	53,520	

>>>

TABLE 3.2 Comparison of Optimus, Opossum, Silvereye, Galapagos (D = 30, NP = 30, termination: 30 min)

		Optimus_jEDE	Opossum_RBFOpt	SilverEye_PSO	Galapagos_GA	Optimal
F_5	*min f(x)*	9.3362001×10^2	4.7527012×10^3	7.5856684×10^3	1.8506721×10^4	-310
	max f(x)	2.5603668×10^3	5.8813877×10^3	1.2910221×10^4	2.4057172×10^4	
	avg f(x)	2.0333032×10^3	5.3824799×10^3	9.4390617×10^3	2.0151105×10^4	
	std f(x)	570.1512256	438.2070353	2031.289127	2004.100331	
	FES	195,720	1891	47,550	67,560	
F_6	*min f(x)*	3.9000000×10^2	1.4168473×10^3	5.0073093×10^2	9.7856127×10^2	390
	max f(x)	3.9398662×10^2	1.3904779×10^4	4.1540000×10^9	9.6995775×10^3	
	avg f(x)	3.9159465×10^2	8.7212119×10^3	9.8402870×10^8	5.6395846×10^3	
	std f(x)	1.9530389×10^0	4.6035484×10^3	1.5972516×10^8	3.5378379×10^3	
	FES	148,260	687	33,540	48,810	
F_7	*min f(x)*	4.5162886×10^3	4.5162896×10^3	5.8669417×10^3	4.5172240×10^3	-180
	max f(x)	4.5162886×10^3	4.5162985×10^3	7.2432580×10^3	4.5290168×10^3	
	avg f(x)	4.5162886×10^3	4.5162936×10^3	6.5251090×10^3	4.5222540×10^3	
	std f(x)	0.0000000×10^0	3.1911420×10^{-3}	5.4380701×10^2	4.7496031×10^0	
	FES	200,820	10,108	42,060	58,290	
F_8	*min f(x)*	-1.1905178×10^2	-1.1910166×10^2	-1.1901775×10^2	-1.1940297×10^2	-140
	max f(x)	-1.1902135×10^2	-1.1876717×10^2	-1.1892500×10^2	-1.1906700×10^2	
	avg f(x)	-1.1903319×10^2	-1.1889866×10^2	-1.1899553×10^2	-1.1919711×10^2	
	std f(x)	1.0538581×10^{-2}	1.1070562×10^{-1}	3.5483336×10^{-2}	1.4127484×10^{-2}	
	FES	149,670	2018	35,580	52,020	
F_9	*min f(x)*	-3.1706554×10^2	-2.5827181×10^2	-2.3690804×10^2	-3.1677970×10^2	-330
	max f(x)	-3.1209075×10^2	-2.3567358×10^2	-1.6682751×10^2	-3.1164785×10^2	
	avg f(x)	-3.1527462×10^2	-2.4625749×10^2	-1.8917798×10^2	-3.1499413×10^2	
	std f(x)	1.7117897×10^0	8.0334497×10^0	2.5285354×10^1	1.8617799×10^0	
	FES	212,160	5577	47,610	67,560	
F_{10}	*min f(x)*	-2.7030257×10^2	-2.3528777×10^2	-1.3299956×10^2	4.0215414×10^1	-330
	max f(x)	-2.2751946×10^2	-1.3298172×10^2	-8.1262211×10^1	1.8543670×10^2	
	avg f(x)	-2.5139841×10^2	-1.8578676×10^2	-1.0572303×10^2	1.1181316×10^2	
	std f(x)	1.4307998×10^1	3.9394042×10^1	1.8528544×10^1	5.7458318×10^1	
	FES	146,820	1192	35,220	53,070	

Results presented in Table 3.2 are provided in supplementary materials containing minimum fitness values of each replication, as well as their corresponding chromosomes. Considering *min f(x)*, the convergence of standard benchmarks and CEC 2005 benchmarks are presented in Figs. 3.8, 3.9, and 3.10.

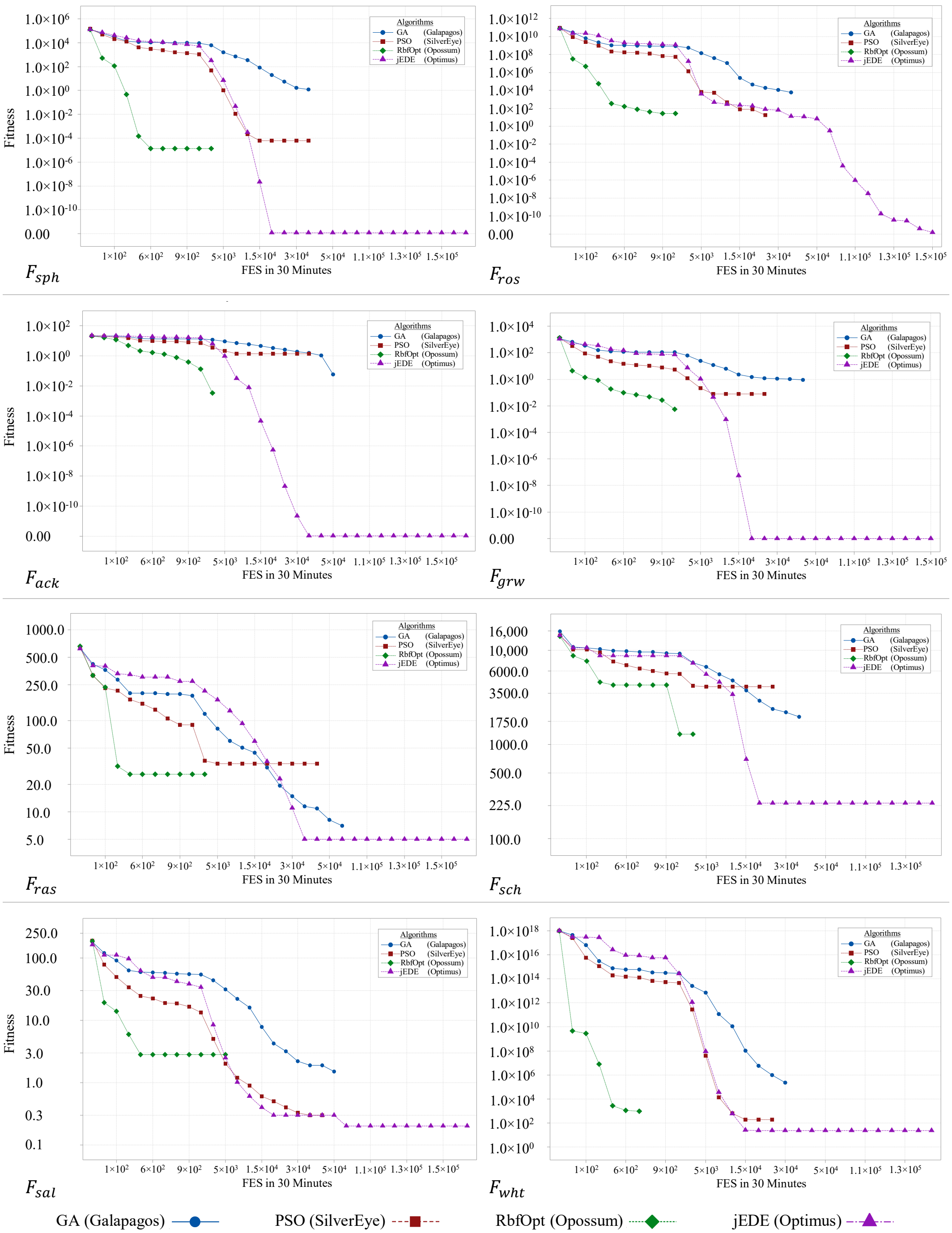


FIG. 3.8 Convergence graphs of standard benchmark functions

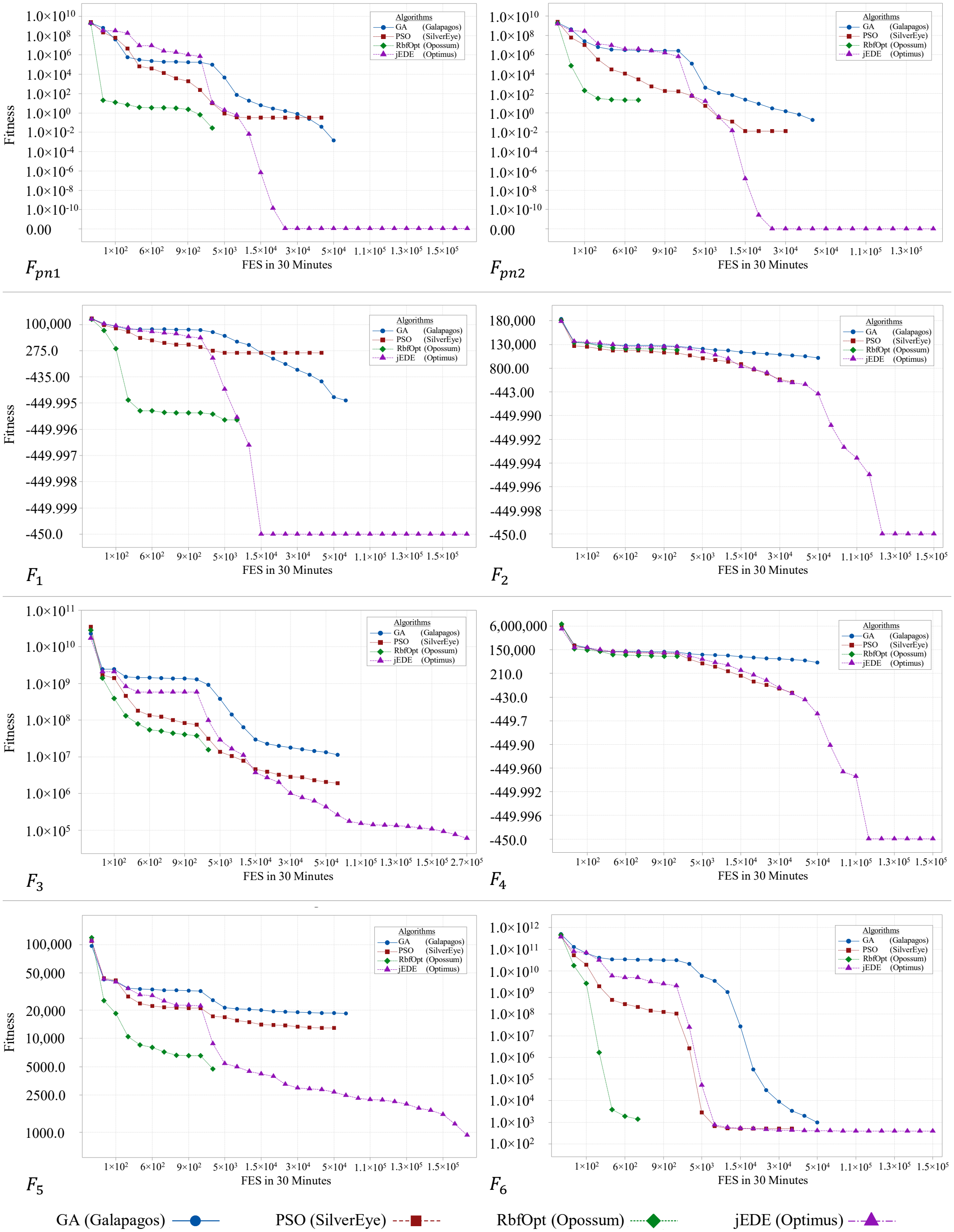


FIG. 3.9 Convergence graphs of standard and CEC2005 benchmark functions

TOC

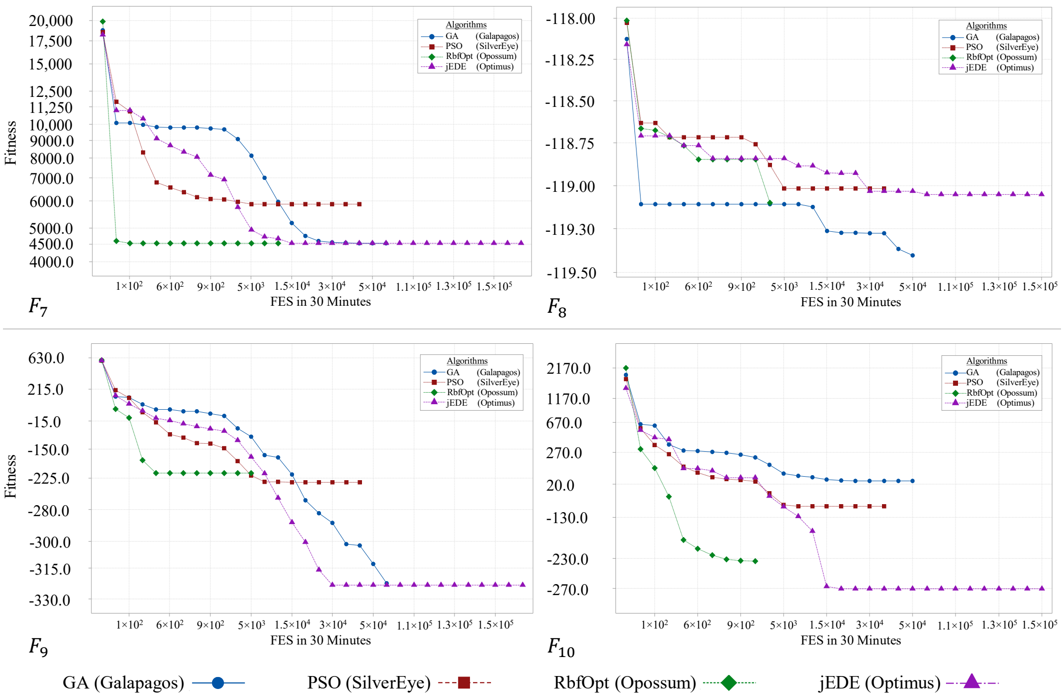


FIG. 3.10 Convergence graphs of CEC2005 benchmark functions

3.4.2 Design optimisation problem

This section presents a design problem for a frame structure, which has a span of 30 m by 25 m. Before generating the frame structure, we executed a base surface, which controls the shape of the design, having 7.5 m height, five axes, and 25 controlling points. Afterwards, 65 parameters are defined for five axes to change the shape of the base surface, so the shape of the frame structure. Points on ground level have two parameters for x and y directions, whereas other points have three parameters for changing the positions in all directions. In the next step, using the base surface, we generated the frame structure with the truss structure component provided by LunchBox [61] plug-in. In this component, the structure on the base surface is controlled by three parameters, which are division amounts on u and v directions and the depth of the truss system. Finally, the generated frame structure is combined with the Karamba 3D plug-in [62], which provides the evaluation of parametric structural models in GH. The structural evaluation process is mentioned in the following lines. The development of the parametric model is illustrated in Fig. 3.11.

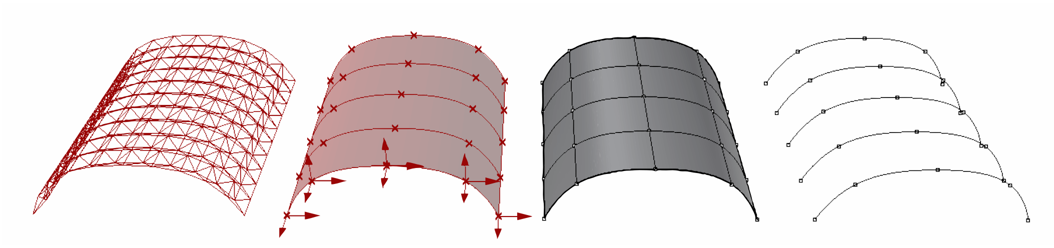


FIG. 3.11 Process of the parametric frame structure

Regarding the performance evaluation of each generated design alternative, each line in the structural model is defined as a beam, whereas each point located on the ground level is defined as a support point. In addition, cross-section and its parameters are also defined by Karamba just before the evaluation process. For the design problem on hand, the rectangular cross-section is defined using three parameters that are the height, upper, and lower widths of the section. Finally, gravity and lateral loads are defined as the last step of the structural evaluation. As a gravity load, the distribution of the total mass for each intersecting point is considered. For the lateral load, 2 kN on each intersecting point is utilised, whereas the material type of the model is assigned as steel. An example of the evaluated model is shown in Fig. 3.12.

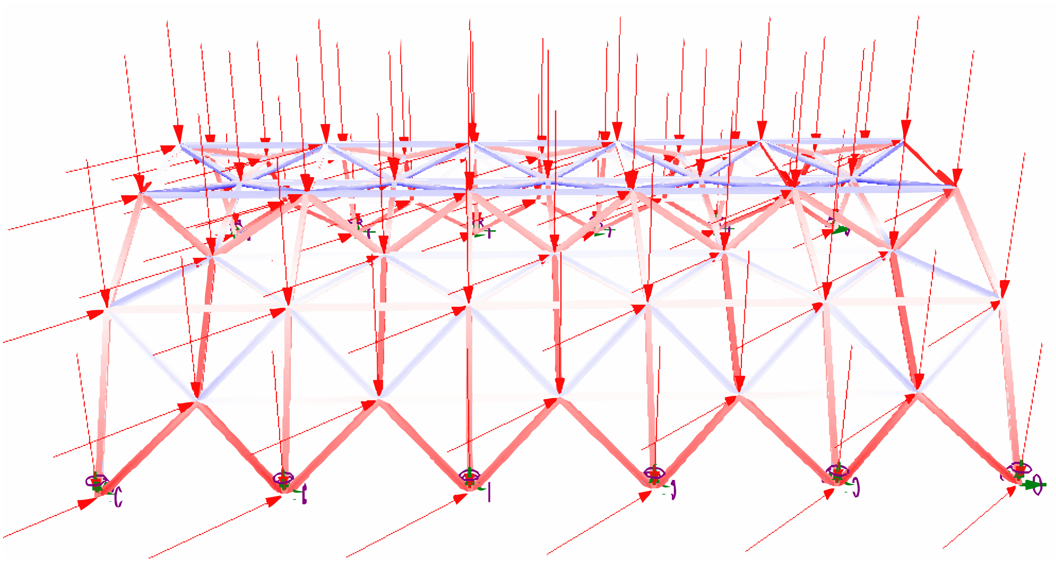


FIG. 3.12 Evaluation of the structure model

To sum up, 70 parameters are used to define the design optimisation problem. This corresponds to approximately 1.333 x 10^{177} alternatives in the search space. When compared to mathematical benchmark problems, design problems may require larger numbers of parameters owing to multiple optimisation tasks that could be involved in the conceptual design phase (e.g., optimising the layout scheme and the façade design of a building to find the most desirable plan scheme with the lowest energy consumption). Therefore, such a large search space with 70 parameters is defined to investigate the performance of the algorithms as a demonstration of a complex design task having large numbers of parameters. Properties of design parameters are given in Table 3.3. Alternatives of the frame structures, as well as the divergence of the design alternatives, are illustrated in Fig. 3.13.

TABLE 3.3 Properties of design parameters

Notation	Design Parameter	Min	Max	Type
x1–x13	Coordinates of control points in axis 1	−2.00	2.00	Continues
x14–x26	Coordinates of control points in axis 2	−2.00	2.00	Continues
x27–x39	Coordinates of control points in axis 3	−2.00	2.00	Continues
x40–x52	Coordinates of control points in axis 4	−2.00	2.00	Continues
x53–x65	Coordinates of control points in axis 5	−2.00	2.00	Continues
x66	Division amount on u direction	3	10	Discrete
x67	Division amount on v direction	3	10	Discrete
x68	Depth of the truss	0.50	1.00	Continues
x69	Height of the cross-section	10.00	30.00	Continues
x70	Upper and lower width of the cross-section	10.00	30.00	Continues

The objective function, which is minimising the mass (m) subject to displacement v, is formulated as follows:

$Min(m)$ where m is given by

$$m = \sum_{i=1}^{j} W_i \ , \tag{3.11}$$

where W_i is the weight of i^{th} element of the frame structure, and j is the total number of the elements in frame structure.

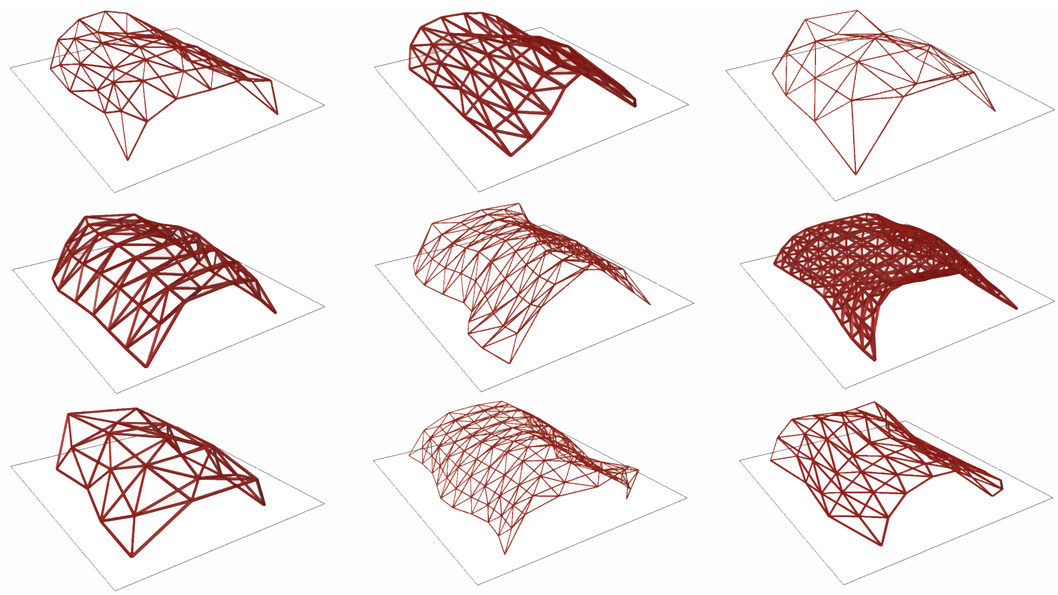


FIG. 3.13 Various alternatives of the design problem

Subject to:

$$v \leq 0.1m\,, \tag{3.12}$$

$$v = \frac{F}{K}\,, \tag{3.13}$$

where F is the loading force, and K is the bending stiffness of the frame structure. To compare the optimisation results of Optimus with others, we defined a penalty function by combining m and v as follows:

$$m = \begin{cases} m & \text{if } v \leq 0.1 \\ 100 \times m & \text{otherwise} \end{cases}. \tag{3.14}$$

For safety reasons, the final design should have a minimum of 0.1m displacement that corresponds to a feasible solution. In addition, for minimising construction cost, the objective is defined as the minimisation of the mass. Hence, the final design should present the smallest amount of steel usage subject to an acceptable value of displacement.

3.4.2.1 Experimental setup and evaluation criteria

The design optimisation problem ran on a computer that had Intel Core I7-6500U CPU @ 2.50 GHz with 16 GB RAM. The number of dimensions D was taken as 70, whereas the population size NP was 50. As such the CEC 2005 benchmark problems, the termination criteria was determined as 30 min for each run. For the design instance on hand, only one replication was carried out for each optimisation tool. To evaluate the performance of different optimisation algorithms, we reported *min f(x)*, which is the minimum fitness value of the design problem, and *g(x)*, which is the constraint value of the minimum fitness. The maximum number of fitness evaluations (FES) within 30 min for each tool were also recorded. The problem definition is available in supplementary materials as GH file to contribute evaluations of further developed architectural design optimisation tools.

3.4.2.2 Design results

After 30 min run for each tool in GH, an overview of optimisation results is given in Table 3.4. The convergence graph for Optimus, Opossum, SilverEye and Galapagos are presented in Fig. 3.14, whereas the final designs proposed by each algorithm are illustrated in Fig. 3.15. The results indicated that Optimus and Opossum found feasible solutions, whereas SilverEye and Galapagos discovered infeasible alternatives. Regarding the feasible results, there was a significant difference between jEDE and RbfOpt as the Optimus found a significantly smaller mass amount than Opossum. During the optimisation process, we also observed that Optimus evaluated more fitness functions than other tools. From the point of proposed (best) design alternatives, Galapagos and SilverEye presented larger frame structures than other algorithms because of smaller profile dimensions. This caused an infeasibility in displacement, thus, undesirable results in mass. On the other hand, Opossum discovered a similar frame size with Galapagos and Silvereye but bigger dimension sizes for the profile parameters. This suggested an acceptable amount of displacement with the usage of 22 tons of steel.

However, the Optimus discovered the smallest mass considering the smallest frame structure and profile size that was the most desirable design alternative for both constraint and fitness functions. Considering a 70-dimensional challenging design problem clearly presented the performance difference when using self-adaptive parameter updates and the ensemble of mutation strategies. Hence, evidence was conducted to employ advanced optimisation algorithms instead of their basic implementations in the architectural design domain. Results presented in Table 3.4 are provided in supplementary materials containing minimum fitness values and their corresponding chromosomes. The best result discovered by Optimus is illustrated in Fig. 3.16.

TABLE 3.4 Comparison of Optimus, Opossum, Silvereye, Galapagos (D = 70, NP = 50, termination: 30 min)

		Optimus_jEDE	Opossum_RBFOpt	SilverEye_PSO	Galapagos_GA
Design problem	*min f(x)*	6.21637×10^3	2.25410×10^4	6.89062×10^5	6.74560×10^5
	g(x)	0.0994	0.0996	0.6674	0.9273
	FES	31,800	5102	17,100	19,000

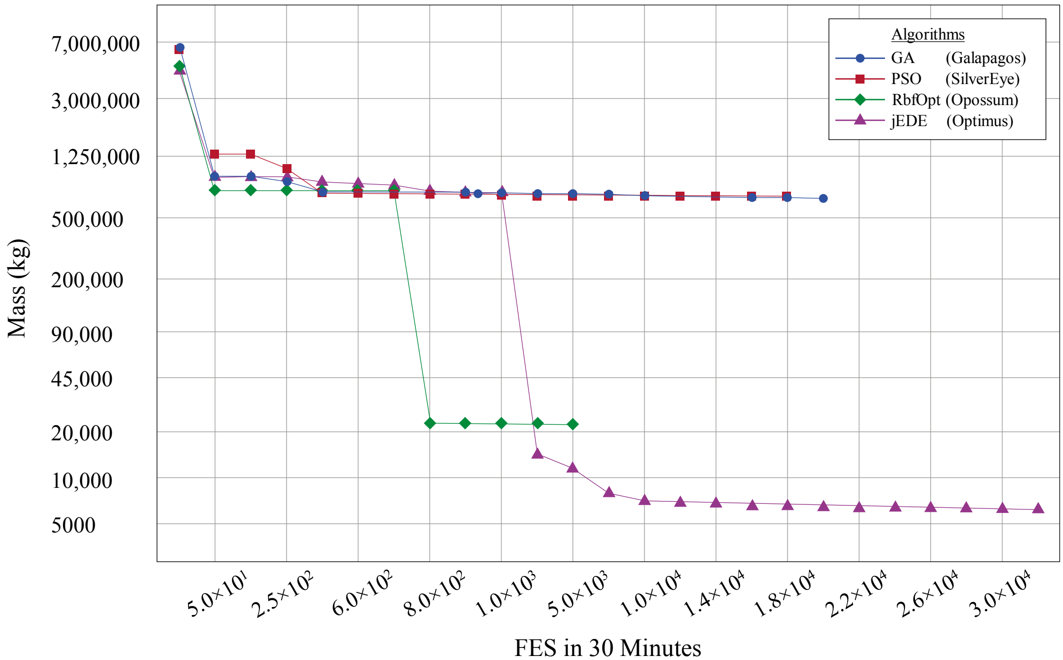


FIG. 3.14 Convergence graph of the architectural design problem

TOC

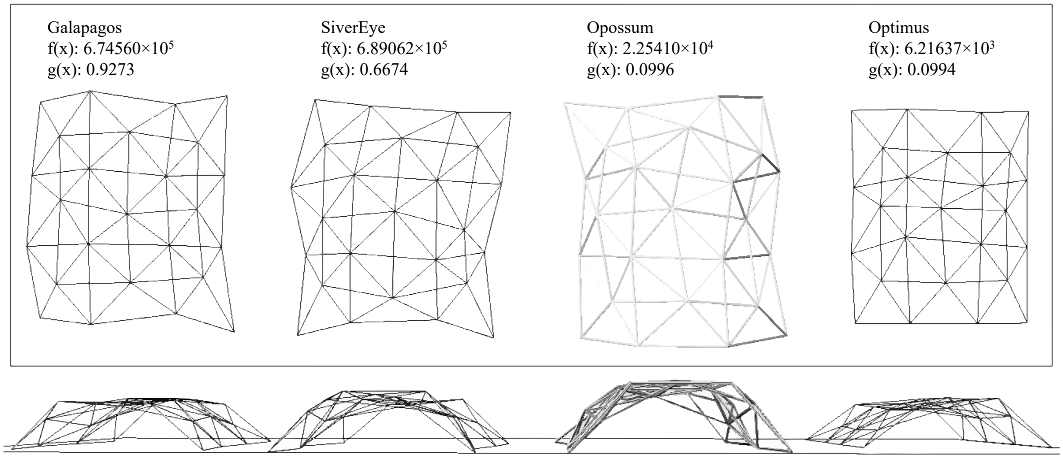


FIG. 3.15 Optimised results of the design problem

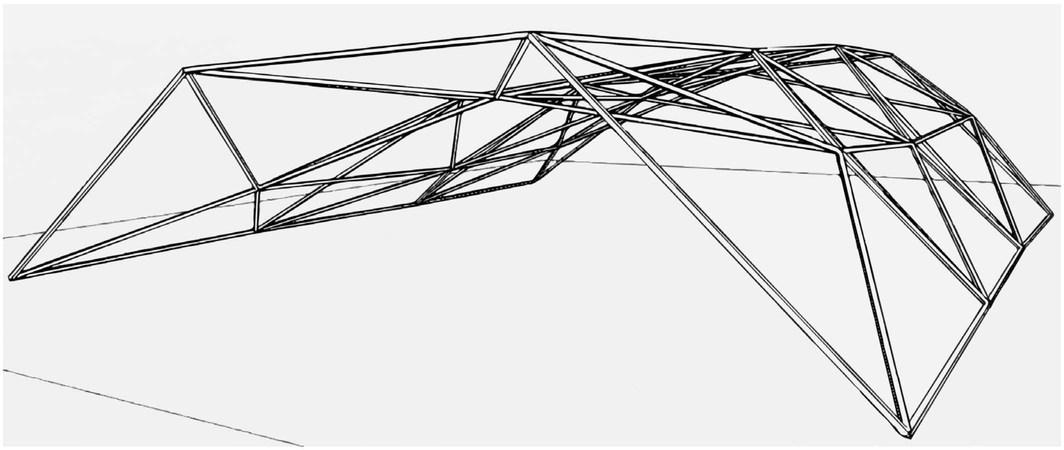


FIG. 3.16 Illustration of the best alternative (Optimus) result for the design optimisation problem

3.5 Discussion

The performance of the Optimus tool is evaluated using 20 benchmark problems and a design optimisation problem to be compared with GA, PSO, RBFOpt. Experimental results showed that jEDE outperforms other algorithms by finding better fitness values. Several reasons can be mentioned to explain this outcome. First, Optimus uses a self-adaptive approach for producing control parameters. This gives the advantage to adapt the search behaviour of the algorithm according to the nature of the problem. Secondly, the developed tool considers the ensemble of mutation strategies, while the other algorithms are based on a single mutation operator. As a result, using more than one mutation strategy enlarged the search space that supports the algorithm for finding better solutions; thus, increasing the ability to search for near-optimal results. Thirdly, Optimus does not update the number sliders of GH, which corresponds to decision variables for each iteration. Instead, generated populations are directly connected to the geometry. Therefore, the required computation time for one generation was less than the other algorithms. Finally, optimal values of some problems are outside of the initialisation range (e.g., F_7). One may argue that this situation can also be the situation in architectural design problems. Generating chromosome values outside of the initialisation range is possible in the Optimus, whereas the other tools have no access to control such procedures.

RBFOpt algorithm is recently compared with metaheuristics using several building optimisation problems in [12–14,32]. According to the results, RBFOpt found better solutions than metaheuristics. In these studies, termination criterion is defined as the number of iterations during the evaluation of different algorithms. This gives an advantage for RBFOpt, due to the ability to discover desirable solutions with a small number of function evaluations. However, metaheuristics require an excessive number of function evaluations to find the optimal solutions. In this study, we defined the termination criterion as run time for 30 min to make a fair comparison. Results indicated that RBFOpt requires more computation time to execute one function than metaheuristics. For this reason, while comparing model-based and metaheuristic algorithms, rather than iterations and/or function evaluations, run time should be considered as a termination criterion.

Fitness functions based on simulations (e.g., energy and daylight) may require an enormous amount of time for convergence during the optimisation. From the point of metaheuristics, the usage of surrogate models [63] can be a solution to overcome this drawback since surrogate models require less amount of function evaluation while

TOC

approximating the fitness. In this way, researchers can consider many replications in a short run time during the optimisation process.

Furthermore, handling constraints is another important topic for architectural design optimisation. Most of the real-world problems require design constraints that may restrict the design alternatives during the optimisation process. There are many constraint handling methods that can be practically integrated into the optimisation algorithms [64]. In the reviewed literature, only Nelder–Mead optimisation tool provides a constraint handling method for the design optimisation problems in the GH. To make a fair comparison with Galapagos, SilverEye, and Opossum, we considered the constant penalty function as a constraint to find a feasible solution in the design optimisation problem. However, there are extremely hard constraints that can be tackled in real-world problems, e.g., equality constraints. Currently, this is not available in GH.

In addition to the advantages of Optimus mentioned above, there are some limitations, as well. Even though the modular approach provides flexibility for using different algorithms, this approach requires more plug-in items to handle in each step of the optimisation process. Another limitation is the evaluation strategy of the fitness function. Optimus creates NP size of the list, where each element in the list has a D size of dimensions. All these parameters (with $NP \times D$ size) are sent to the objective function for fitness evaluation. Afterwards, NP size of fitness results is obtained simultaneously. This procedure gives an advantage for Optimus in terms of computation time, but it may not be suitable for some of the architectural design problems owing to the more actions to be taken during the function evaluation.

3.6 Conclusion

In conclusion, this paper presented a new optimisation plug-in called Optimus for GH algorithmic modelling environment in the Rhinoceros CAD program. A self-adaptive differential evolution algorithm with an ensemble of mutation strategies was implemented in Optimus for coping with the complex architectural design problems. An experimental design was introduced by using standard test problems in the literature, some of the test problems proposed in IEEE CEC 2005 and an architectural design problem to evaluate the proposed algorithm's performance. In general, Optimus outperformed other optimisation algorithms available in GH. The paper also showed that an algorithm that presents a promising performance in solving real-parameter benchmark functions could also find desirable alternatives in solving architectural design problems. Therefore, the paper aims to support decision-makers while coping with large numbers of design parameters in the domain of architectural design optimisation through the presented tool.

As future work, Optimus will be further improved by implementing different types of metaheuristic algorithms due to the NFL theorem. These algorithms can be variants of PSO, GA, HS, and DE. Moreover, Optimus can be updated for constrained optimisation problems using near-feasibility threshold [65,66], Superior of Feasibility [67], and Epsilon constraint [68]. An ensemble of constraint handling techniques [69] can also be used in Optimus that may play a crucial role in constrained design problems. Finally, Optimus can be extended for the multi-objective optimisation domain owing to its modular system.

Appendix 3A. Benchmark functions

F_{sph} : *Sphere Function* $F_{sph}(x)=\sum_{i=1}^{D} x_i^2$

$F_{sph}^*(x^*)=(0,..,0)=0 \quad -100 \le x \le 100$

F_{ros} : *Rosenbrock's Function* $F_{ros}(x)=\sum_{i=1}^{D-1}\left(100\left(x_{i+1}-x_i^2\right)^2+\left(1-x_i\right)^2\right)$

$F_{ros}^*(x^*)=(1,..,1)=0 \quad -100 \le x \le 100$

F_{ack} : *Ackley's Function* $F_{ack}(x)=-20\exp\left(-0.2\sqrt{\frac{1}{D}\sum_{i=1}^{D}x_i^2}\right)-\exp\left(\frac{1}{D}\sum_{i=1}^{D}\cos\left(2\pi x_i\right)\right)+20+e$

$F_{ack}^*(x^*)=(0,..,0)=0 \quad -32 \le x \le 32$

F_{grw} : *Griewank's Function* $F_{grw}(x)=\sum_{i=1}^{D}\frac{x_i^2}{4000}-\prod_{i=1}^{D}\cos\left(\frac{x_i}{\sqrt{i}}\right)+1$

$F_{grw}^*(x^*)=(0,..,0)=0 \quad -600 \le x \le 600$

F_{ras} : *Rastrigin's Function* $F_{ras}(x)=\sum_{i=1}^{D}\left(x_i^2-10\cos\left(2\pi x_i\right)+10\right)$

$F_{ras}^*(x^*)=(0,..,0)=0 \quad -5 \le x \le 5$

F_{sch} : *Generalised Schwefel's Problem 2.26* $F_{sch}(x)=418.9829N-\sum_{i=1}^{D}\left(x_i\sin\left(\left|x_i\right|\right)\right)$

$F_{sch}^*(x^*)=(420.9687,..,420.968)=0 \quad -500 \le x \le 500$

F_{sal} : *Salomon's Function* $F_{sal}(x)=-\cos\left(2\pi\sqrt{\sum_{i=1}^{D}x_i^2}\right)+0.1\sqrt{\sum_{i=1}^{D}x_i^2}+1$

$F_{sal}^*(x^*)=(0,..,0)=0 \quad -100 \le x \le 100$

F_{wht} : *Whitely's Function* $F_{wht}(x) = \sum_{j=1}^{D}\sum_{i=1}^{D}\left(\frac{y_{ij}^2}{4000} - \cos(y_{ij}) + 1\right)$

where $y_{ij} = 100(x_j - x_i)^2 + (1 - x_i)^2$

$F_{wht}^*(x^*) = (1,..,1) = 0 \quad -100 \le x \le 100$

F_{pn1} : *Generalised Penalised Function 1*

$$F_{pn1}(x) = \frac{\pi}{D}\left\{10\sin^2(\pi y_1) + \sum_{i=1}^{D-1}(y_i - 1)^2\left[1 + 10\sin^2(\pi y_{i+1})\right] + (y_D - 1)^2\right\} + \sum_{i=1}^{D} u(x_i, 10, 100, 4)$$

where $y_i = 1 + \frac{1}{4}(x_i + 1)$ and $u(x_i, a, k, m) = \begin{cases} k(x_i - a)^m & x_i > a \\ 0 & -a \le x_i \le a \\ k(-x_i - a)^m & x_i < a \end{cases}$

$F_{pn1}^*(x^*) = (-1,..,-1) = 0 \quad -50 \le x \le 50$

F_{pn2} : *Generalised Penalised Function 2*

$$F_{pn2}(x^*) = 0.1\left\{\sin^2(3\pi x_1) + \sum_{i=1}^{D-1}(x_{i-1} - 1)^2\left[1 + \sin^2(3\pi x_{i+1})\right] + (x_D - 1)^2\left[1 + \sin^2(2\pi x_D)\right]\right\} + \sum_{i=1}^{D} u(x_i, 5, 100, 4)$$

$F_{pn2}^*(x^*) = (1,..,1) = 0 \quad -50 \le x \le 50$

F_1 : *Shifted Sphere Function* $F_1(x) = \sum_{i=1}^{D} z_i^2 + f\_bias_1$

$x \in [-100, 100]^D$, $F_1(x^*) = f\_bias_1 = -450$

F_2 : *Shifted Schwefel's Problem 1.2* $F_2(x) = \sum_{i=1}^{D}\left(\sum_{j=1}^{i} z_j\right)^2 + f\_bias_2$

$x \in [-100, 100]^D$, $F_2(x^*) = f\_bias_2 = -450$

F_3 : *Shifted Rotated High Conditioned Elliptic Function* $F_3(x) = \sum_{i=1}^{D}(10^6)^{\frac{i-1}{D-1}} z_i^2 + f\_bias_3$

$x \in [-100, 100]^D$, $F_3(x^*) = f\_bias_3 = -450$

F_4 *: Shifted Schwefel's Problem 1.2 with Noise in Fitness*

$$F_4(x) = \left(\sum_{i=1}^{D}\left(\sum_{j=1}^{i} z_i\right)^2\right) \times \left(1 + 0.4\left|N(0,1)\right|\right) + f\_bias_4$$

$$x \in [-100,100]^D,\ F_4(x^*) = f\_bias_4 = -450$$

F_5 *: Schwefel's Problem 2.6 with Global Optimum on Bounds*

$$F_5(x) = \max\{|A_i x - B_i|\} + f\_bias_5, i = 1,2,..,D$$

$$x \in [-100,100]^D,\ F_5(x^*) = f\_bias_5 = -310$$

F_6 *: Shifted Rosenbrock's Function* $F_6(x) = \sum_{i=1}^{D-1}\left(100\left(z_i^2 - z_{i+1}\right)^2 + (z_i - 1)^2\right) + f\_bias_6$

$$x \in [-100,100]^D,\ F_4(x^*) = f\_bias_6 = 390$$

F_7 *: Shifted Rotated Griewank's Function without Bounds*

$$F_7(x) = \sum_{i=1}^{D}\frac{z_i^2}{4000} - \prod_{i=1}^{D}\cos\left(\frac{z_i}{\sqrt{i}}\right) + 1 + f\_bias_7$$

Initialise population in $[0,600]^D$, Global optimum is outside of the initialisation range,
$F_7(x^*) = f\_bias_7 = -180$

F_8 *: Shifted Rotated Ackley's Function with Global Optimum on Bounds.*

$$F_8(x) = -20\exp\left(-0.2\sqrt{\frac{1}{D}\sum_{i=1}^{D} z_i^2}\right) - \exp\left(\frac{1}{D}\sum_{i=1}^{D}\cos(2\pi z)\right) + 20 + e + f\_bias_8$$

$$x \in [-32,32]^D,\ F_8(x^*) = f\_bias_8 = -140$$

F_9 *: Shifted Rastrigin's Function* $F_9(x) = \sum_{i=1}^{D}\left(z_i^2 - 10\cos(2\pi z_i) + 10\right) + f\_bias_9$

$$x \in [-5,5]^D,\ F_9(x^*) = f\_bias_9 = -330$$

F_{10} *: Shifted Rotated Rastrigin's Function* $F_{10}(x) = \sum_{i=1}^{D}\left(z_i^2 - 10\cos(2\pi z_i) + 10\right) + f\_bias_{10}$

$$x \in [-5,5]^D,\ F_{10}(x^*) = f\_bias_{10} = -330$$

References

[1] Sariyildiz, S. Performative Computational Design, Keynote Speech. In Proceedings of the ICONARCH-I: International Congress of Architecture-I, Konya, Turkey, 15–17 November **2012**.

[2] Ekici, B.; Cubukcuoglu, C.; Turrin, M.; Sariyildiz, I.S. Performative computational architecture using swarm and evolutionary optimisation: A review. *Build. Environ.* **2018**, *147*, 356–371.

[3] Michalewicz, Z.; Fogel, D.B. *How to Solve it: Modern Heuristics*; Springer Science & Business Media: New York, NY, USA, **2013**.

[4] Geem, Z.W.; Kim, J.H.; Loganathan, G.V. A new heuristic optimization algorithm: Harmony search. *Simulation* **2001**, *76*, 60–68.

[5] Eberhart, R.; Kennedy, J. A New Optimizer Using Particle Swarm Theory. In Proceedings of the MHS'95. Sixth International Symposium on Micro Machine and Human Science, Nagoya, Japan, 4–6 October **1995**; pp. 39–43.

[6] Storn, R. On the Usage of Differential Evolution for Function Optimization. In Proceedings of the North American Fuzzy Information Processing, Berkeley, CA, USA, 19–22 June **1996**; pp. 519–523.

[7] Storn, R.; Price, K. Differential evolution—A simple and efficient heuristic for global optimization over continuous spaces. *J. Glob. Optim.* **1997**, *11*, 341–359.

[8] Goldberg, D.E.; Holland, J.H. Genetic algorithms and machine learning. *Mach. Learn.* **1988**, *3*, 95–99.

[9] Dorigo, M.; Gambardella, L.M. Ant colony system: A cooperative learning approach to the traveling salesman problem. *IEEE Trans. Evol. Comput.* **1997**, *1*, 53–66.

[10] Kirkpatrick, S.; Gelatt, C.D.; Vecchi, M.P. Optimization by simulated annealing. *Science* **1983**, *220*, 671–680.

[11] Coello, C.A.C.; Lamont, G.B.; van Veldhuizen, D.A. *Evolutionary Algorithms for Solving Multi-Objective Problems*; Springer: Berlin/Heidelberg, Germany, **2007**; Volume 5.

[12] Wortmann, T. Genetic Evolution vs. Function Approximation: Benchmarking Algorithms for Architectural Design Optimization. *J. Comput. Des. Eng.* **2018**, *6*, 414–428.

[13] Wortmann, T.; Waibel, C.; Nannicini, G.; Evins, R.; Schroepfer, T.; Carmeliet, J. Are Genetic Algorithms Really the Best Choice for Building Energy Optimization? In Proceedings of the Symposium on Simulation for Architecture and Urban Design, Toronto, CA, Canada, 22–24 May **2017**; p. 6.

[14] Waibel, C.; Wortmann, T.; Evins, R.; Carmeliet, J. Building energy optimization: An extensive benchmark of global search algorithms. *Energy Build.* **2019**, *187*, 218–240.

[15] Cichocka, J.M.; Migalska, A.; Browne, W.N.; Rodriguez, E. SILVEREYE—The Implementation of Particle Swarm Optimization Algorithm in a Design Optimization Tool. In Proceedings of the International Conference on Computer-Aided Architectural Design Futures, Istanbul, Turkey, 10–14 July **2017**; pp. 151–169.

[16] Rutten, D. Galapagos: On the logic and limitations of generic solvers. *Archit. Des.* **2013**, *83*, 132–135.

[17] Camporeale, P.E.; del Moyano, P.M.M.; Czajkowski, J.D. Multi-objective optimisation model: A housing block retrofit in Seville. *Energy Build.* **2017**, *153*, 476–484.

[18] Calcerano, F.; Martinelli, L. Numerical optimisation through dynamic simulation of the position of trees around a stand-alone building to reduce cooling energy consumption. *Energy Build.* **2016**, *112*, 234–243.

[19] Anton, I.; Tãnase, D. Informed geometries. Parametric modelling and energy analysis in early stages of design. *Energy Procedia* **2016**, *85*, 9–16.

[20] Tabadkani, A.; Banihashemi, S.; Hosseini, M.R. Daylighting and visual comfort of oriental sun responsive skins: A parametric analysis. *Build. Simul.* **2018**, *11*, 663–676.

[21] Lee, K.; Han, K.; Lee, J. Feasibility study on parametric optimization of daylighting in building shading design. *Sustainability* **2016**, *8*, 1220.

[22] Fathy, F.; Sabry, H.; Faggal, A.A. External Versus Internal Solar Screen: Simulation Analysis for Optimal Daylighting and Energy Savings in an Office Space. In Proceedings of the PLEA, Edinburgh, UK, 16 August **2017**.

[23] Lavin, C.; Fiorito, F. Optimization of an external perforated screen for improved daylighting and thermal performance of an office space. *Procedia Eng.* **2017**, *180*, 571–581.

[24] Heidenreich, C.; Ruth, J. Parametric Optimization of Lightweight Structures, 11th World Congress on Computational Mechanics, **2014**.

[25] Eisenbach, P.; Grohmann, M.; Rumpf, M.; Hauser, S. Seamless Rigid Connections of Thin Concrete Shells—A Novel Stop-End Construction Technique for Prefab Elements. In Proceedings of IASS Annual Symposia, Amsterdam, The Netherlands, 17–20 August **2015**, pp. 1–12.

[26] Almaraz, A. Evolutionary optimization of parametric structures: Understanding structure and architecture as a whole from early design stages, Master's thesis, University of Coruna, **2015**.
[27] Simon. Goat. **2013**. Available online: https://www.food4rhino.com/app/goat (accessed on: 10 July 2019).
[28] Johnson, S.G. The Nlopt Nonlinear-Optimization Package. Available online: https://nlopt.readthedocs.io/en/latest/ (accessed on: 10 July 2019).
[29] Ilunga, G.; Leitão, A. Derivative-free Methods for Structural Optimization. In Proceedings of the 36th eCAADe Conference, Lodz, Poland, 19–21 September **2018**.
[30] Austern, G.; Capeluto, I.G.; Grobman, Y.J. Rationalization and Optimization of Concrete Façade Panels. In Proceedings of the 36th eCAADe Conference, Lodz, Poland, 19–21 September **2018**.
[31] Delmas, A.; Donn, M.; Grosdemouge, V.; Musy, M.; Garde, F. Towards Context & Climate Sensitive Urban Design: An Integrated Simulation and Parametric Design Approach. In Proceedings of the 4th International Conference On Building Energy & Environment 2018 (COBEE2018), Melbourne, Australia, 5–9 February **2018**.
[32] Wortmann, T. Opossum: Introducing and Evaluating a Model-based Optimization Tool for Grasshopper. In Proceedings of the CAADRIA 2017, Hong Kong, China, 5–8 July **2017**.
[33] Costa, A.; Nannicini, G. RBFOpt: An open-source library for black-box optimization with costly function evaluations. *Math. Program. Comput.* **2018**, *10*, 597–629.
[34] Wortmann, T. Model-based Optimization for Architectural Design: Optimizing Daylight and Glare in Grasshopper. *Technol. Archit. Des.* **2017**, *1*, 176–185.
[35] Greco, L. Dodo. **2015**. Available online: https://www.food4rhino.com/app/dodo (accessed on: 10 July 2019).
[36] Eckersley O'Callaghan's Digital Design Group. **2013**. Nelder-Mead Optimization. Available online: https://www.food4rhino.com/app/nelder-mead-optimisation-eoc (accessed on: 10 July 2019).
[37] Lagarias, J.C.; Reeds, J.A.; Wright, M.H.; Wright, P.E. Convergence properties of the Nelder-Mead simplex method in low dimensions. *SIAM J. Optim.* **1998**, *9*, 112–147.
[38] Wrenn, G.A. *An Indirect Method for Numerical Optimization Using the Kreisselmeir-Steinhauser Function*; NASA: Washington, DC, USA, **1989**.
[39] Wolpert, D.H.; Macready, W.G. No free lunch theorems for optimization. *IEEE Trans. Evol. Comput.* **1997**, *1*, 67–82.
[40] Attia, S.; Hamdy, M.; O'Brien, W.; Carlucci, S. Assessing gaps and needs for integrating building performance optimization tools in net zero energy buildings design. *Energy Build.* **2013**, *60*, 110–124.
[41] Robert McNeel & Associates. Rhinoceros 3D. NURBS Modelling. **2015**. Available online: https://www.rhino3d.com/ (accessed on: 10 July 2019).
[42] Brest, J.; Greiner, S.; Boskovic, B.; Mernik, M.; Zumer, V. Self-adapting control parameters in differential evolution: A comparative study on numerical benchmark problems. *IEEE Trans. Evol. Comput.* **2006**, *10*, 646–657.
[43] Mallipeddi, R.; Suganthan, P.N.; Pan, Q.-K.; Tasgetiren, M.F. Differential evolution algorithm with ensemble of parameters and mutation strategies. *Appl. Soft Comput.* **2011**, *11*, 1679–1696.
[44] Suganthan, P.N.; Hansen, N.; Liang, J.J.; Deb, K.; Chen, Y.-P.; Auger, A.; Tiwari, S. Problem definitions and evaluation criteria for the CEC 2005 special session on real-parameter optimization. *KanGAL Rep.* **2005**.
[45] Blackwell, T.M.; Kennedy, J.; Poli, R. Particle swarm optimization. *Swarm Intell.* **2007**, *1*, 33–57.
[46] Sengupta, S.; Basak, S.; Peters, R. Particle Swarm Optimization: A survey of historical and recent developments with hybridization perspectives. *Mach. Learn. Knowl. Extr.* **2018**, *1*, 157–191.
[47] Mirjalili, S. Genetic Algorithm. In *Evolutionary Algorithms and Neural Networks*; Springer: New York, NY, USA, **2019**; pp. 43–55.
[48] Koza, J.R. *Genetic Programming: On the Programming of Computers by Means of Natural Selection*; MIT press: Cambridge, MA, USA, **1992**; Volume 1.
[49] Knowles, J.; Corne, D. The Pareto Archived Evolution Strategy: A New Baseline Algorithm for Pareto Multiobjective Optimisation. In Proceedings of the Congress on Evolutionary Computation (CEC99), Washington, DC, USA, 6–9 July **1999**; Volume 1, pp. 98–105.
[50] Pan, Q.-K.; Tasgetiren, M.F.; Liang, Y.-C. A discrete differential evolution algorithm for the permutation flowshop scheduling problem. *Comput. Ind. Eng.* **2008**, *55*, 795–816.
[51] Venu, M.K.; Mallipeddi, R.; Suganthan, P.N. Fiber Bragg grating sensor array interrogation using differential evolution. *Optoelectron. Adv. Mater. Commun.* **2008**, *2*, 682–685.
[52] Varadarajan, M.; Swarup, K.S. Differential evolution approach for optimal reactive power dispatch. *Appl. Soft Comput.* **2008**, *8*, 1549–1561.

[53] Das, S.; Konar, A. Automatic image pixel clustering with an improved differential evolution. *Appl. Soft Comput.* **2009**, *9*, 226–236.
[54] Chatzikonstantinou, I.; Ekici, B.; Sariyildiz, I.S.; Koyunbaba, B.K. Multi-Objective Diagrid Façade Optimization Using Differential Evolution. In Proceedings of the 2015 IEEE Congress on Evolutionary Computation (CEC), Sendai, Japan, 25–28 May **2015**; pp. 2311–2318.
[55] Cubukcuoglu, C.; Chatzikonstantinou, I.; Tasgetiren, M.F.; Sariyildiz, I.S.; Pan, Q.-K. A multi-objective harmony search algorithm for sustainable design of floating settlements. *Algorithms* **2016**, *9*, 51.
[56] Das, S.; Suganthan, P.N. Differential evolution: A survey of the state-of-the-art. *IEEE Trans. Evol. Comput.* **2011**, *15*, 4–31.
[57] Das, S.; Mullick, S.S.; Suganthan, P.N. Recent advances in differential evolution—an updated survey. *Swarm Evol. Comput.* **2016**, *27*, 1–30.
[58] Tasgetiren, M.F.; Suganthan, P.N.; Pan, Q.-K.; Mallipeddi, R.; Sarman, S. An Ensemble of Differential Evolution Algorithms for Constrained Function Optimization. In Proceedings of the IEEE congress on evolutionary computation, Vancouver, BC, Canada, 24–29 July **2016**; pp. 1–8.
[59] Hansen, N.; Ostermeier, A. Completely derandomized self-adaptation in evolution strategies. *Evol. Comput.* **2001**, *9*, 159–195.
[60] Chatzikonstantinou, I. HoopSnake. **2012**. Available online: https://www.food4rhino.com/app/hoopsnake (accessed on: 10 July 2019).
[61] Miller, N. LunchBox. **2012**. Available online: https://www.food4rhino.com/app/lunchbox (accessed on: 10 July 2019).
[62] Preisinger, C.; Heimrath, M. Karamba—A toolkit for parametric structural design. *Struct. Eng. Int.* **2014**, *24*, 217–221.
[63] Shan, S.; Wang, G.G. Survey of modeling and optimization strategies to solve high-dimensional design problems with computationally-expensive black-box functions. *Struct. Multidiscip. Optim.* **2010**, *41*, 219–241.
[64] Coello, C.A.C. Theoretical and numerical constraint-handling techniques used with evolutionary algorithms: A survey of the state of the art. *Comput. Methods Appl. Mech. Eng.* **2002**, *191*, 1245–1287.
[65] Tasgetiren, M.F.; Suganthan, P.N. A Multi-Populated Differential Evolution Algorithm for Solving Constrained Optimization Problem. In Proceedings of the 2006 IEEE International Conference on Evolutionary Computation, Vancouver, BC, Canada, 16–21 July **2006**; pp. 33–40.
[66] Coit, D.W.; Smith, A.E. Penalty guided genetic search for reliability design optimization. *Comput. Ind. Eng.* **1996**, *30*, 895–904.
[67] Deb, K. An efficient constraint handling method for genetic algorithms. *Comput. Methods Appl. Mech. Eng.* **2000**, *186*, 311–338.
[68] Haimes, Y.V. On a bicriterion formulation of the problems of integrated system identification and system optimization. *IEEE Trans. Syst. Man. Cybern.* **1971**, *1*, 296–297.
[69] Mallipeddi, R.; Suganthan, P.N. Ensemble of constraint handling techniques. *IEEE Trans. Evol. Comput.* **2010**, *14*, 561–579.

INTRODUCTION

	METHOD(S)	CHALLENGE(S)	
LITERATURE REVIEW			
J1 **RQ1**	Optimising form-finding parameters in performative computational architecture	■ 17 Form-finding parameters ● 13 Performance aspects ▲ 12 Evolutionary algorithms 3 Swarm algorithms	
TOOL DEVELOPMENT AND PILOT STUDY			
J2 **RQ2**	PART A Developing Optimus tool using self-adaptive ensemble evolutionary algorithm	■ 30 Parameters ▲ 4 Optimisation algorithms	TEST 1
		■ 70 Parameters ▲ 4 Optimisation algorithms	TEST 2
J3 **RQ3**	PART B Preliminary results of multi-zone approach using pilot high-rise model	■ 100 Parameters ● 2 Daylight metrics ▲ 5 ANN models 1 Optimisation algorithm	TEST 3
METHODOLOGICAL FRAMEWORK			
J4 **RQ4**	PART A Introducing multi-zone optimisation (MUZO) methodology and prediction results of quad-grid and diagrid high-rise scenarios	■ 260 Parameters ● 2 Daylight metrics ▲ 20 ANN models	TEST 4
		■ 220 Parameters ● 2 Daylight metrics ▲ 20 ANN models	TEST 5
J5 **RQ5**	PART B Optimising high-rise scenarios using the predictive models with Optimus and validation of the MUZO methodology	■ 260 Parameters ● 20 Predictive models ▲ 3 Optimisation algorithms	TEST 4
		■ 220 Parameters ● 20 Predictive models ▲ 3 Optimisation algorithms	TEST 5
CASE STUDY			
J6 **RQ6**	Optimising Europoint complex for self-sufficiency in energy consumption and food production using MUZO and Optimus	■ 117 Parameters ● 1 Self-sufficiency in energy 1 Self-sufficiency in food 1 Daylight metric ▲ 45 ANN models 13 Optimisation algorithms	

CONCLUSIONS

J Journal
RQ Research question

■ Parameter ● Performance ▲ AI Method

TOC

PART B

A Methodology for daylight optimisation of high-rise buildings in the dense urban district using overhang length and glazing type variables with surrogate modelling

ABSTRACT

Urbanisation and population growth lead to the construction of higher buildings in the 21st century. This causes an increment in energy consumption as the amount of constructed floor areas is rising steadily. Integrating daylight performance in building design supports reducing energy consumption and satisfying occupants' comfort. This study presents a methodology to optimise the daylight performance of a high-rise building located in a dense urban district. The purpose is to deal with optimisation problems by dividing the high-rise building into five zones from the ground level to the sky level, to achieve better daylight performance. Therefore, the study covers five optimisation problems. Overhang length and glazing type are considered to optimise spatial Daylight Autonomy (sDA) and Annual Sunlight Exposure (ASE). A total of 500 samples in each zone are collected to develop surrogate models. A self-adaptive differential evolution algorithm is used to obtain near-optimal results for each zone. The developed surrogate models can estimate the metrics with a minimum of 98.25% R^2, which is calculated from neural network prediction and Diva simulations. In the case study, the proposed methodology improves daylight performance of the high-rise building, decreasing ASE by approx. 27.6% and increasing the sDA values by around 88.2% in the dense urban district.

3.7 Introduction: Pilot model

High-rise buildings have been designed to gain additional floor area in the limited urban plot since the early examples [1]. In the 21st century, population growth and a trend towards urbanisation lead construction of higher buildings increasingly. Owing to a rise in constructed spaces, this suggests an increment in the energy consumption to meet the requirements for thermal and visual comfort [2]. In this respect, daylight becomes an important performance aspect for high-rises because designing spaces with good daylight performance helps reducing energy consumption and satisfy occupants' comfort requirements. However, this is a complex task owing to design decisions given in the conceptual phase. First, many design parameters such as the shape of the building, design of the shading devices, and material properties suggest an enormous number of design alternatives affecting the building performance. Thus, finding a desirable set of parameters during the decision is very challenging in the early phases. Secondly, daylight requirements can vary relevantly depending on the indoor functions, which are often mixed in high-rises. Thirdly, in several climates, the need for daylight conflicts with the need of reducing indoor solar loads. Finally, a possible design solution cannot be applied at all heights of the high-rise. In fact, due to the surrounding buildings in the dense districts, optimal design parameters for good daylight performance can be different, starting from the ground level to the sky level.

Very limited studies can be found for daylight optimisation of high-rises. One study [3] focuses on proposing modifications to extend the daylight deeper into the space using extra interior height, alternative glazing and an external light shelf for a commercial high-rise building. In another study [4], authors presented a holistic passive design approach to evaluate a typical high-rise residential building focusing on daylight, natural ventilation and thermal comfort. Recently, researchers [5] considered a simulation-based multi-objective optimisation to minimise energy loads, reduce CO_2 emissions, and improve occupants' health and comfort for high-rise and low-rise buildings. All these studies present promising results and conclusions. However, none of these studies considered different design parameter sets for different parts of the high-rises that can further improve the daylight performance. This study presents a methodology to optimise daylight performance of a whole high-rise building located in a dense urban district, considering a variety of parameter sets at different subdivisions (zones) of the building.

TOC

3.8 Methodology

Daylight availability in the upper zones of the high-rise building, which are close to the sky, is different from daylight availability in the lower zones near the ground level since the surrounding buildings cause obstruction on the façade in dense urban environments. Such a situation may result in specific requirements on daylight performances at each floor/zone level in the building. For instance, an optimised parameter set near the sky level may not perform desirable performance solutions for the ground level and the other way around. Thus, the idea of the proposed methodology is based on a holistic approach, which aims to consider each corresponding zone of high-rise buildings as an optimisation problem. In this respect, desirable parameter sets for each zone/level can be tested and evaluated. There are four steps which are proposed in methodology (Fig. 3.17). These are:

- Form finding: A parametric model of a high-rise building is generated defining design parameters. Five equally divided zones are defined (zone 1-5) for the performance assessment.
- Performance evaluation: Corresponding floors are selected for daylight simulation in all zones.
- Surrogate modelling: Uniformly generated samples are collected for each part to define fitness function and constraint using surrogate modelling based on artificial neural networks (ANN).
- Optimisation: The most desirable parameter sets in each zone are discovered using a computational optimisation algorithm.

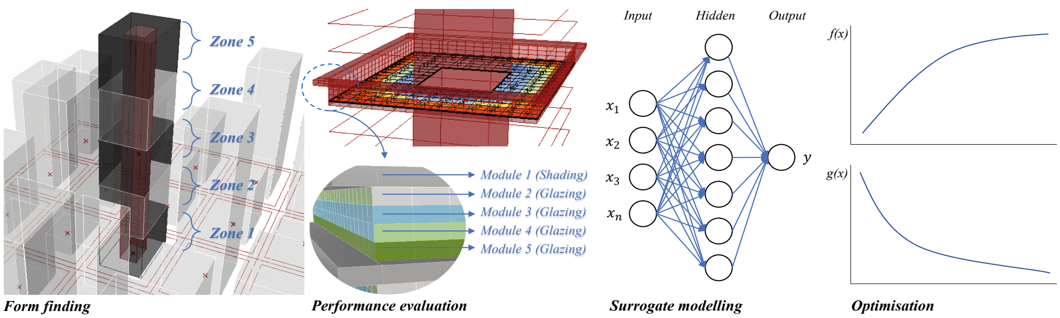


FIG. 3.17 Schematic explanation of proposed methodology

TOC

3.8.1 Form-finding

A hypothetical urban district with 25 plots is generated in Grasshopper 3d (GH) [6]. Each plot has 1800 m^2 with randomly generated heights from 50 m to 150 m. The central plot is defined as the case area having 40 floors, 200 m height, 72000 m^2 area, and 36 to 50 m façade length. The generated building is divided into five zones named Zone-1 (Z1), Zone-2 (Z2), Zone-3 (Z3), Zone-4 (Z4), and Zone-5 (Z5). The floors in the middle part of each zone are selected for the simulation. The façade is divided into five vertical modules. The first four modules are defined as glazing. The last module is defined as an overhang. It is possible to assign four types of glazing material to every four modules of each orientation, whereas the length of each overhang can vary from 0 m to 2 m. Parameters with boundaries are given in Table 3.5. The design alternatives of 20 variables are 47.223665e+20.

TABLE 3.5 Design parameters

Parameters	Explanation	Type	Boundary
$x_1,\ldots,x_4$	Glazing type for North (N) orientation	Discrete	[1,4]
$x_5,\ldots,x_8$	Glazing type for South (S) orientation		[1,4]
$x_9,\ldots,x_{12}$	Glazing type for East (E) orientation		[1,4]
$x_{13},\ldots,x_{16}$	Glazing type for West (W) orientation		[1,4]
$x_{17},\ldots,x_{20}$	Overhang length for N-S-E-W orientations	Continues	[0.0, 2.0]

3.8.2 Performance evaluation

Spatial Daylight Autonomy (sDA) and Annual Sunlight Exposure (ASE) are considered to assess the daylight performance in each zone. According to the Illumination Engineering Society (IES) [7], sDA is a metric for sufficient daylight illuminance, whereas ASE is a metric for the potential visual discomfort owing to direct sunlight. More specifically, sDA calculates the percentage of an analysis area that meets with the minimum illuminance level for a specified operating hour per year. ASE calculates the percentage of an analysis area that exceeds a specified direct sunlight illuminance level more than a specified number of hours per year. Diva plug-in [8] in GH is used to simulate these metrics. An analysis plane, which is 0.8m above the finished floor with 184 sensors, is generated. $sDA_{300,50\%}$, which achieves the illumination threshold of 300 lux for 50% of the analysis period, is considered. $ASE_{1000,250h}$, which exposes the illumination threshold of 1000 lux for 250 hours of the analysis period, is used. For both metrics, 10 hours (8 am-6 pm) is specified.

Glazing types (Table 3.6) are assigned to all orientations in sequence. Radiance parameters of the daylight simulation are given in Table 3.7. One simulation is recorded as 103.9 seconds with the given radiance parameters.

TABLE 3.6 Material characterisation of glazing types

Material	Explanation	Tvis.	U-val.	g-val.
Glazing 1 (G1)	Tinted Float 8mm Blue – 12 mm Air – Temperable Low-E 8mm Blue	0.22	1.6	28%
Glazing 2 (G2)	Temperable Low-E 8mm Neutral – 12 mm Air – Clear Float 8 mm – 12 mm Air – Temperable Low-E 8 mm Green	0.45	0.9	40%
Glazing 3 (G3)	Tinted Float 8 mm Green	0.68	5.6	51%
Glazing 4 (G4)	Ultra-clear Float 8 mm – 12 mm Cavity Air – Ultra-clear Float 8 mm	0.82	2.8	81%

TABLE 3.7 Radiance parameters

-aa	-ab	-ad	-ar	-as
0.15	2	512	256	128

3.8.3 Surrogate modelling

Five hundred samples are collected for each zone to develop the surrogate models. A uniform distribution function, coded in C#, is used to generate random values. Every recorded data contains 20 design parameters and simulation results for sDA and ASE. In total, 2500 samples are collected for five different zones in 72.1 hours. Implementing simulation results of these 500 design samples, ANN models are developed using a Backpropagation neural network algorithm with bipolar sigmoid activation function using Dodo plug-in [9]. After several experiments, 20 input, 1 hidden, and 1 output layer are considered. To sum up, five ANN models for sDA and five models for ASE are developed. sDA and ASE values of collected samples are given in Table 3.8. These predicted values (outputs) obtained from ANN are compared to outputs from Diva simulations with similar design parameters, calculating the R^2 values of each model (Table 3.9). This procedure shows us the applicability of the ANN model.

TABLE 3.8 sDA (%) and ASE (%) distributions for each zone

	Z1-sDA	Z1-ASE	Z2-sDA	Z2-ASE	Z3-sDA	Z3-ASE	Z4-sDA	Z4-ASE	Z5-sDA	Z5-ASE
Min	80.6	25.7	85.8	31.5	87.2	33.3	84.2	33.8	88.5	32.1
Max	100.0	49.6	100.0	58.4	100.0	58.4	100.0	66.5	100.0	66.5
Avg	95.9	42.6	98.2	47.0	98.8	47.2	99.0	49.2	99.1	49.6

TABLE 3.9 Parameters and R-squares of ANN models

	Neurons per layer	Number of layers	Learning rate	Sigmoid alpha	Max iter	R^2-Z1	R^2-Z2	R^2-Z3	R^2-Z4	R^2-Z5
sDA	20	1	0.1	2.0	10,000	98.25%	99.43%	99.26%	99.43%	99.85%
ASE				0.5		99.95%	99.74%	99.82%	99.94%	98.95%

3.8.4 Optimisation

Subsequently, the optimisation problem is formulated as follows:

$$\max(sDA_{300,50\%}), \tag{3.15}$$

subject to

$$ASE_{1000,250h} \leq 20\% . \tag{3.16}$$

The single-objective self-adaptive differential evolution (jDE) algorithm [10], coded in C#, is used for optimisation. The implementation is based on DE/rand/1/bin scheme, which uses three individuals to generate the mutant population, and employs one-to-one comparison for the next generation. Rather than constant mutation (MR) and crossover (CR) rates, in jDE, these values are updated for each individual in D dimensions during the optimisation. In addition, to cope with the constraint, the superior-of-feasibility (SF) procedure [11] is implemented. SF considers three cases, which are: Pick the solution with better fitness value, pick the feasible solution, or pick the solution with smaller violation.

3.9 Results

Average values of initial and 500th generation with a population size of 30 are conducted for each zone. Simulation results for regular building cases using only one glazing material without overhang were conducted to prove the relevance of the proposed methodology (Table 3.10). The convergence of the optimisation process is presented in Fig. 3.18. Until the 250th generation, ASE values decreased, whereas, during the last 100 generations, sDA values did not present a significant alteration. From the point of comparison between initial and optimised populations, the average of optimised results reached a minimum value of 27.6% smaller ASE than initial results. However, this caused a maximum value of 10.3% decrement on sDA. When we compare the optimised and regular (G1 to G4) results, the proposed methodology found significantly smaller ASE values than each case.

In general, most of the optimised values presented a maximum value of 11.8% decrement on sDA. Finally, optimised parameters were applied to corresponding zones to finalise the design of the high-rise (Fig. 3.19). Since the optimised ASE range was very narrow for all zones, results having the highest sDA values were picked. The colours of materials were defined as blue for G1, light green for G2, dark green for G3, white for G4, and grey for overhangs. In the optimised high-rise building, the total usage amount of glazing material was 47.5% for G1, 17.5% for G2, G3, and G4. Average overhang distances were reported as 1.4m in Z1, 1.5m in Z2, 2.0m in Z3, 1.2m in Z4, and 1.1m in Z5. It was observed that overhang distances and material selections were differentiated in all zones and orientations to find the best near-optimal solution.

TABLE 3.10 Results for initial, optimised, and regular building cases

	Init. sDA			Init. ASE			Opt. sDA			Opt. ASE			G1 (0.22)		G2 (0.45)		G3 (0.68)		G4 (0.82)	
	min	max	avg	min	max	avg	min	max	avg	min	max	avg	sDA	ASE	sDA	ASE	sDA	ASE	sDA	ASE
Z1	84.5	100	95.1	26.1	47.7	36.8	87.6	89.2	**88.2**	23.6	23.6	**23.6**	63.0	37.9	95.3	49.0	100	49.0	100	49.6
Z2	93.4	100	98.4	36.1	49.2	44.2	88.0	88.6	**88.3**	29.2	29.2	**29.2**	64.8	46.7	98.8	49.0	100	50.2	100	58.4
Z3	95.6	100	99.2	35.6	48.4	43.0	88.6	94.3	**90.0**	32.2	31.2	**31.2**	65.4	46.7	100	49.0	100	51.4	100	58.4
Z4	94.2	100	98.9	37.3	54.3	44.1	96.6	97.4	**97.0**	31.1	31.1	**31.1**	66.0	49.0	99.4	49.6	100	58.4	100	67.7
Z5	89.9	100	98.4	35.9	52.5	44.3	88.1	100	**97.6**	29.1	29.5	**29.3**	66.6	49.0	99.4	49.6	100	58.4	100	67.7

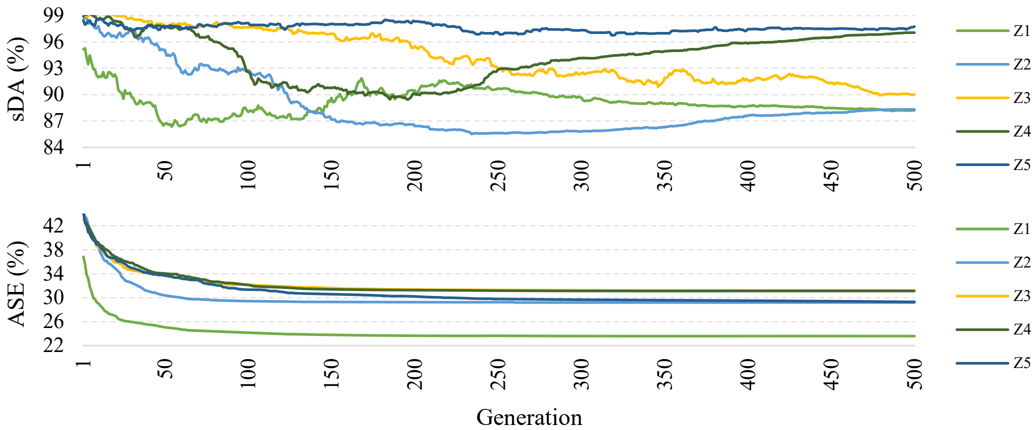


FIG. 3.18 Average sDA and ASE values during the optimisation

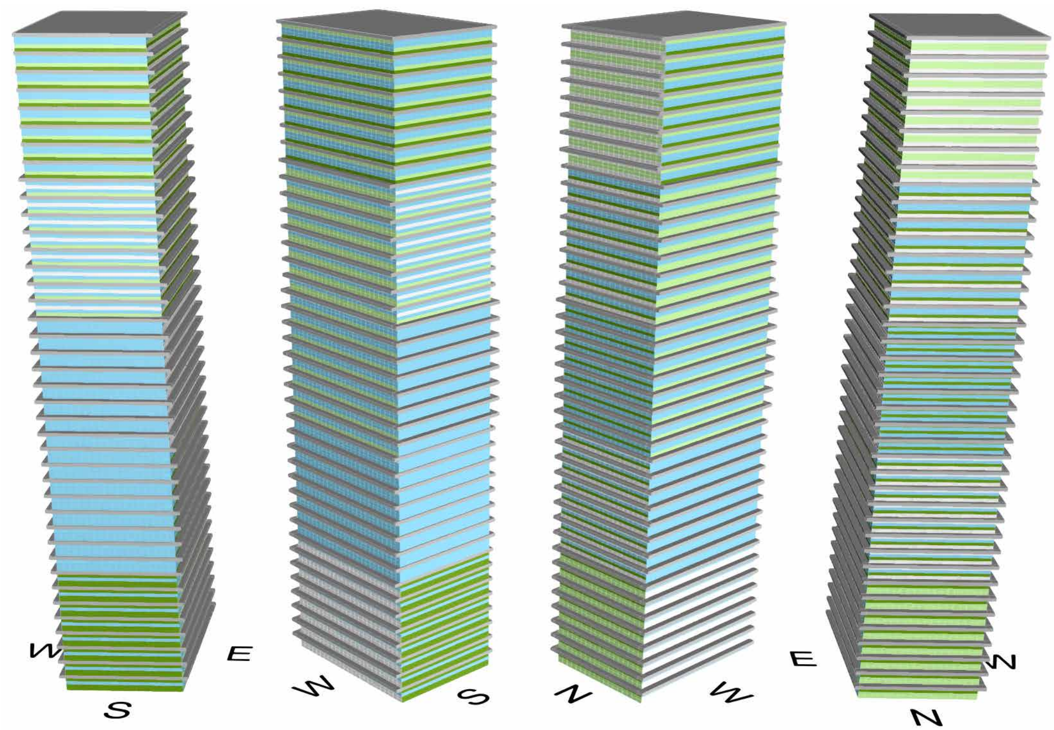


FIG. 3.19 Schematic distribution of optimised parameters

3.10 Discussions and conclusion

This paper presents a methodology to optimise sDA and ASE for high-rise buildings in dense urban districts. Surrogate models successfully approximated metrics with a minimum value of 98.25% R^2 when predicted ANN outputs are compared to simulation outputs. In the case of simulation-based optimisation, the required time for metaheuristics would correspond to one simulation time for 15,000 function evaluations. Using ANN with 500 samples for each zone, we saved approximately 90 days to conduct the presented results. The number of samples can exceed thousands with more design parameters. In this case, an additional optimisation process would be necessary to find the best architecture and parameters for ANN. Here, the results of the initial population and regular building cases were compared with the optimised solution. The proposed methodology clearly showed that the daylight performance of the high-rise building was improved in all zones. The minimum enhancement for ASE was 27.6% in Z3, whereas the maximum advancement was 35.9% in Z1. sDA was reported in the acceptable margins between 88.2% and 97.6%, indicating spaces successfully benefiting from daylight. Although optimised solutions were not checked against thermal performance, higher ASE values (23-31%) than required draw our attention to a potential of overheating in these cases. So, this study can be an initial step to suggest further research for testing decrements on thermal energy consumption of such high-rise buildings in temperate-humid climates. Thus, zones at varying levels of high-rise buildings require combinations of parameter sets to perform the best solution in this sense. Specifically, infeasible ASE values remind us of the necessity of a shading approach once again.

References

[1] Gottmann, J. J. G. r., Why the Skyscraper? **1966**, 190-212.

[2] Ali, M. M.; Al-Kodmany, K. J. B., Tall buildings and urban habitat of the 21st century: a global perspective. **2012,** 2, (4), 384-423.

[3] Close, J. J. R. E., Optimising daylighting in high-rise commercial developments in SE Asia and the use of computer programmes as a design tool. **1996,** 8, (1-4), 206-209.

[4] Chen, X.; Yang, H.; Sun, K. J. E., A holistic passive design approach to optimize indoor environmental quality of a typical residential building in Hong Kong. **2016,** 113, 267-281.

[5] Matic, D.; Calzada, J.; Todorovic, M.; Erić, M.; Babin, M., Cost-Effective Energy Refurbishment of Prefabricated Buildings in Serbia. In *Cost-Effective Energy Efficient Building Retrofitting*, Elsevier: **2017**; pp 455-487.

[6] Rutten, D.; McNeel, R., Grasshopper Algorithmic Modeling for Rhino, www.grasshopper3d.com. **2012**.

[7] IES, LM-83-12: Approved Method: IES Spatial Daylight Autonomy (sDA) and Annual Sunlight Exposure (ASE). In New York: Illuminating Engineering Society of North America: **2012**.

[8] Jakubiec, J. A.; Reinhart, C. F. In *DIVA 2.0: Integrating daylight and thermal simulations using Rhinoceros 3D, Daysim and EnergyPlus*, Proceedings of building simulation, Citeseer: **2011**; pp 2202-2209.

[9] Greco, L., Dodo, https://www.food4rhino.com/app/dodo. **2015**.

[10] Brest, J.; Zumer, V.; Maucec, M. S. In *Self-adaptive differential evolution algorithm in constrained real-parameter optimization*, IEEE international conference on evolutionary computation, **2006**; pp 215-222.

[11] Deb, K.; Pratap, A.; Agarwal, S.; Meyarivan, A fast and elitist multiobjective genetic algorithm: NSGA-II. In IEEE Transactions on Evolutionary Computation, **2002,** 6, (2), 182-197.

INTRODUCTION	METHOD(S)	CHALLENGE(S)

LITERATURE REVIEW

J1 RQ1	Optimising form-finding parameters in performative computational architecture	■ 17 Form-finding parameters ● 13 Performance aspects ▲ 12 Evolutionary algorithms 3 Swarm algorithms

TOOL DEVELOPMENT AND PILOT STUDY

J2 RQ2	PART A Developing Optimus tool using self-adaptive ensemble evolutionary algorithm	■ 30 Parameters ▲ 4 Optimisation algorithms	TEST 1
		■ 70 Parameters ▲ 4 Optimisation algorithms	TEST 2
J3 RQ3	PART B Preliminary results of multi-zone approach using pilot high-rise model	■ 100 Parameters ● 2 Daylight metrics ▲ 5 ANN models 1 Optimisation algorithm	TEST 3

METHODOLOGICAL FRAMEWORK

J4 RQ4	PART A Introducing multi-zone optimisation (MUZO) methodology and prediction results of quad-grid and diagrid high-rise scenarios	■ 260 Parameters ● 2 Daylight metrics ▲ 20 ANN models	TEST 4
		■ 220 Parameters ● 2 Daylight metrics ▲ 20 ANN models	TEST 5
J5 RQ5	PART B Optimising high-rise scenarios using the predictive models with Optimus and validation of the MUZO methodology	■ 260 Parameters ● 20 Predictive models ▲ 3 Optimisation algorithms	TEST 4
		■ 220 Parameters ● 20 Predictive models ▲ 3 Optimisation algorithms	TEST 5

CASE STUDY

J6 RQ6	Optimising Europoint complex for self-sufficiency in energy consumption and food production using MUZO and Optimus	■ 117 Parameters ● 1 Self-sufficiency in energy 1 Self-sufficiency in food 1 Daylight metric ▲ 45 ANN models 13 Optimisation algorithms

CONCLUSIONS

J Journal
RQ Research question

■ Parameter ● Performance ▲ AI Method

TOC

4 Multi-zone optimisation (MUZO) methodology

Part A of Chapter 4 has been published as: Ekici, B.; Kazanasmaz, Z. T.; Turrin, M.; Taşgetiren, M. F.; Sariyildiz, I. S., Multi-zone optimisation of high-rise buildings using artificial intelligence for sustainable metropolises. Part 1: Background, methodology, setup, and machine learning results. Solar Energy 2021, 224, 373-389.
Part B of Chapter 4 has been published as: Ekici, B.; Kazanasmaz, Z. T.; Turrin, M.; Taşgetiren, M. F.; Sariyildiz, I. S., Multi-zone optimisation of high-rise buildings using artificial intelligence for sustainable metropolises. Part 2: Optimisation problems, algorithms, results, and method validation. Solar Energy 2021, 224, 309-326.
For consistency of the dissertation, the layout is adapted to fit the template, some typos are adjusted, and phrases are reworded without changing the content.

Part A: https://doi.org/10.1016/j.solener.2021.05.083
Part B: https://doi.org/10.1016/j.solener.2021.05.082

This chapter presents the multi-zone optimisation (MUZO) methodology that entails the parametric high-rise model, machine learning for surrogate models, computational optimisation, and decision-making. Part A of this chapter presents the entire methodology and two design scenarios indicated as Tests 4 and 5 to demonstrate the relevance of the MUZO. Both scenarios, focusing on quad-grid and diagrid façade designs, integrate frequently used form-finding parameters for building shape and façade design reported in Chapter 2. Additionally, Part A conducts the machine learning results using the parametric high-rise models to cope with the computationally expensive simulation time while assessing the performance of the entire building. Afterwards, Part B presents the optimisation problems and results of both design scenarios using the predictive models developed in Part A and the released version of the Optimus tool presented in Chapter 3. Since the study focuses on optimising the entire design of the high-rise scenarios are considered 260 and 220 design parameters, respectively, for quad-grid and diagrid scenarios. Consequently, Part B presents the relevance of the MUZO methodology by comparing the results with the regular high-rise scenarios, which use the same design parameters in the entire building.

	METHOD(S)	CHALLENGE(S)	
INTRODUCTION			
LITERATURE REVIEW			
J1 RQ1	Optimising form-finding parameters in performative computational architecture	■ 17 Form-finding parameters ● 13 Performance aspects ▲ 12 Evolutionary algorithms 3 Swarm algorithms	
TOOL DEVELOPMENT AND PILOT STUDY			
J2 RQ2	PART A Developing Optimus tool using self-adaptive ensemble evolutionary algorithm	■ 30 Parameters ▲ 4 Optimisation algorithms	TEST 1
		■ 70 Parameters ▲ 4 Optimisation algorithms	TEST 2
J3 RQ3	PART B Preliminary results of multi-zone approach using pilot high-rise model	■ 100 Parameters ● 2 Daylight metrics ▲ 5 ANN models 1 Optimisation algorithm	TEST 3
METHODOLOGICAL FRAMEWORK			
J4 RQ4	PART A Introducing multi-zone optimisation (MUZO) methodology and prediction results of quad-grid and diagrid high-rise scenarios	■ 260 Parameters ● 2 Daylight metrics ▲ 20 ANN models	TEST 4
		■ 220 Parameters ● 2 Daylight metrics ▲ 20 ANN models	TEST 5
J5 RQ5	PART B Optimising high-rise scenarios using the predictive models with Optimus and validation of the MUZO methodology	■ 260 Parameters ● 20 Predictive models ▲ 3 Optimisation algorithms	TEST 4
		■ 220 Parameters ● 20 Predictive models ▲ 3 Optimisation algorithms	TEST 5
CASE STUDY			
J6 RQ6	Optimising Europoint complex for self-sufficiency in energy consumption and food production using MUZO and Optimus	■ 117 Parameters ● 1 Self-sufficiency in energy 1 Self-sufficiency in food 1 Daylight metric ▲ 45 ANN models 13 Optimisation algorithms	
CONCLUSIONS			

J Journal
RQ Research question

■ Parameter ● Performance ▲ AI Method

PART A

Multi-zone optimisation of high-rise buildings using artificial intelligence for sustainable metropolises

PART 1 – Background, methodology, setup, and machine learning results

ABSTRACT

Designing high-rise buildings is one of the complex tasks of architecture because it involves interdisciplinary performance aspects in the conceptual phase. The necessity for sustainable high-rise buildings has increased owing to the demand for metropolises based on population growth and urbanisation trends. Although artificial intelligence (AI) techniques support swift decision-making when addressing multiple performance aspects related to sustainable buildings, previous studies only examined single floors because modelling and optimising the entire building requires extensive computational time. However, different floor levels require various design decisions because of the performance variances between the ground and sky levels of high-rises in dense urban districts. This paper presents a multi-zone optimisation (MUZO) methodology to support decision-making for an entire high-rise building considering multiple floor levels and performance aspects. The proposed methodology includes parametric modelling and simulations of high-rise buildings, as well as machine learning and optimisation as AI methods. The specific setup focuses on the quad-grid and diagrid shading devices using two daylight metrics of LEED: spatial daylight autonomy and annual sunlight exposure. The parametric model generated samples to develop surrogate models using an artificial neural network. The results of 40 surrogate models indicated that the machine learning part of the MUZO methodology can report very high prediction accuracies for 31 models and high accuracies for six quad-grid and three diagrid models. The findings indicate that the MUZO can be an important part of designing high-rises in metropolises while predicting multiple performance aspects related to sustainable buildings during the conceptual design phase.

KEYWORDS

Performance-based design, building simulation, sustainability, high-rise building, machine learning, optimisation

4.1 Introduction: Part A

High-rise buildings began to emerge at the end of the 19th century to provide extra floor space in limited urban plots [1]. In the 20th century, population growth and urbanisation trends increased in the world [2]. According to a United Nations report [3], 30% of the world's population lived in urban areas in 1950. This percentage increased to 55% in 2018, and the projection by 2050 was 68%. An increase in population and the percentage of those living in urban areas will add 2.5 billion people to the world's urban population by 2050. Moreover, there were 33 megacities with more than 10 million inhabitants in 2018. The projection indicates that this number will increase to 43 by 2030. Because of population growth and urbanisation trends, the number and height of completed high-rise buildings have also increased over time [4].

Owing to a rapid and global increase in floor areas, the final energy use of buildings reached approximately 128 exajoules (EJ) in 2019, while it was 118 EJ in 2010 [5]. An increasing number of high-rise buildings contribute significantly to energy use as they consume more energy with an additional effect of CO_2 emissions compared with low-rise buildings [6]. Another consequence of constructing more and taller high-rise buildings is the increase in building density in urban areas [7]. To achieve the targets of the International Energy Agency for sustainable development scenarios, architects and engineers should consider the following challenges while designing high-rise buildings for metropolises:

- Dense urban areas cause performance variations between ground and sky levels in high-rise buildings [8].
- Sustainable buildings require the integration of multiple performance aspects, such as natural daylight, energy consumption, and comfort [9].

During the design process, the conceptual phase requires a high awareness of decisions because it affects the overall performance of the buildings [10]. Owing to the complexity of design problems, optimisation algorithms are widely used to investigate sustainable design alternatives during the conceptual design phase [11]. Because the performance aspects of sustainable buildings require simulations, the optimisation process entails a significant amount of time. The common approach is to integrate machine learning (ML) techniques to predict performance aspects to support swift decision-making with optimisation algorithms in computationally expensive design problems [12]. Optimising high-rise buildings in dense urban districts is more challenging because various floor levels require different design decisions owing to performance variations in ground and sky levels. In addition, these decisions are based on simulations, which require expensive computational time, and optimisation processes that need to cope with an enormous number of design parameters. Therefore, new methods are required to optimise the multiple floor levels of high-rise buildings when proposing sustainable alternatives within a limited time.

This paper introduces a novel multi-zone optimisation (MUZO) methodology of optimising high-rises by considering multiple floor levels as different optimisation problems to investigate sustainable alternatives during the conceptual phase. The proposed methodology includes parametric modelling and simulations of high-rise buildings, an artificial neural network (ANN) (an ML technique based on a network of neurones) for performance prediction, and computational optimisation with a decision framework. Part 1 of the MUZO study focuses on solving computationally expensive simulations while presenting the background, methodology, and setup for case studies that contain two types of shading devices: quad-grid and diagrid. The building performance model focuses on the daylight metrics of Leadership in Energy and Environmental Design (LEED) v4.1, namely, spatial daylight autonomy (sDA) and annual sunlight exposure (ASE), for each scenario. The results present the learning scores of 40 surrogate models developed for each performance aspect using advanced ANN techniques. Part 2 of the MUZO study deals with the optimisation challenge while explaining the problem formulations to optimise the sDA and ASE using the 40 predictive models presented in this paper. Considering the near feasibility threshold adaptive penalty function, the optimisation process employs three algorithms, namely, self-adaptive differential evolution with the ensemble of mutation strategies using the Optimus plug-in [13], covariance matrix adaptation with evolution strategy, and radial basis function optimisation using the Opossum plug-in [14]. After validating the method by comparing the MUZO results with the regular high-rise scenarios, the paper discusses the advantages and disadvantages and underlines the potential and future research directions. In this paper, Section 4.2 presents state of the art, Section 4.3 introduces the MUZO methodology, Section 4.4 explains the setup, Section 4.5 reports the ANN results, and Section 4.6 concludes the paper.

4.2 State of the art for AI in the design of sustainable high-rises

This section presents previous studies focusing on performance aspects related to sustainable high-rise buildings in three subsections: ML, computational optimisation, and ML with optimisation applications. Subsequently, the original contribution of the MUZO methodology is summarised.

4.2.1 Machine learning applications

Over the last two decades, ML techniques have been used to address the computational burden of simulations of high-rise buildings. An early study discussed regression models to predict energy performance [15]. After a decade, Ko, et al. [16] focussed on the daylight factor as part of the LEED v2.2 for building shape, layout, and façade parameters. In the following years, Li and Li [17] examined the annual ventilation rate, in addition to energy performance. Tian, et al. [18] developed models for energy-efficient heating design in office buildings considering conventional modelling processes and an innovative two-step method. Recently, researchers began to use sensitivity analyses with ML techniques to decrease the design complexity [19,20]. Since the early years, various aspects have been used to predict the performance of high-rises. However, none of these studies focussed on predicting the performance of an entire building. The general approach focussed on a single-floor level (or part) of the high-rise model.

4.2.2 Computational optimisation applications

High-rise buildings are one of the complex design tasks of architecture because various decisions are required for the shape, layout, and façade parameters considering multiple performance aspects. Therefore, different methods have been examined to address the complexity of these buildings. Considering resource production systems, Imam and Kolarevic [21] proposed a concept to optimise energy, food, water, and land in high-rises. In addition to producing energy, two studies focussed on energy performance [22,23], one study examined the energy demand with adaptive thermal comfort [24], and another considered techno-economic aspects [25].

In addition to façade parameters, Gan, et al. [26] investigated the geometric, position, and functional attributes to optimise energy efficiency. Despite the promising results and design alternatives, two studies [17,23] considered the surroundings of the plots being studied, one study compared various optimisation algorithms [22], and none of them replicated the heuristic optimisation process. Consequently, the general approach, as in ML applications, focuses on a single floor level (or part) of the high-rise model.

4.2.3 Machine learning and computational optimisation applications

In the high-rise domain, early examples of predictive models focussed on evaluating the design performances. Some of the recent studies considered predictive models with optimisation algorithms because of the potential to determine optimal solutions in a short time. Early examples used regression models, support vector machines (SVMs), and multi-objective optimisation [27,28]. In addition, Chen, et al. [29] conducted a sensitivity analysis to decrease the design complexity. Despite the fast evaluation potential of using ML with optimisation, the aforementioned studies considered specific floor levels of high-rise models.

4.2.4 Original contribution of the research

The MUZO methodology is proposed to optimise the entire shape of a high-rise building to investigate sustainable design alternatives while addressing the computational burden. Because dense urban areas result in performance variations between the ground and sky levels, a unique optimisation strategy is required. Therefore, the MUZO methodology suggests dividing the high-rise building into equal subdivisions (or zones), which can be considered as different design problems. In addition, this paper suggests an advanced model selection to provide high prediction accuracies, as well as a decision framework by comparing the algorithms and replicating the optimisation process. Thus, the MUZO methodology aims to determine the optimal design solution by achieving sustainable high-rise alternatives for dense urban districts.

4.3 Multi-zone optimisation methodology

Previous studies demonstrated that the optimisation of high-rise buildings can focus on multiple performance aspects that may require various digital platforms. Considering the flexibility of integrating different software, Fig. 4.1 shows the phases of the MUZO methodology. The parametric high-rise model focuses on generating design alternatives with performance evaluations in phase 1. ML for surrogate models addresses the computational burden of multiple performance aspects related to sustainable buildings in phase 2. Finally, the computational optimisation and decision-making phase investigates the desirable performance for the entire high-rise building.

4.3.1 Parametric high-rise model

The first phase of the methodology considers the parametric model, which involves generating configurations of the high-rise building using design variables. Preparing the model requires three steps: developing the parametric high-rise model to generate design alternatives, identifying zones according to the surroundings of the plot being studied, and integrating performance aspects.

Step 1 (Generating high-rise alternatives)

Initially, creating the context around the plot area is the first step during the development of the parametric high-rise model. This is because surroundings with different densities may require various design strategies and parameters in the conceptual phase. When the built environment is modelled, a parametric high-rise model that involves decision variables related to the building shape, façade design, layout, and operation is generated. Few tools are available for use in this step, i.e. Generative Components [30,31], Dynamo [32,33], and Grasshopper 3D (GH) [34]. The MUZO methodology can include all of these parameter types and available tools during form generation.

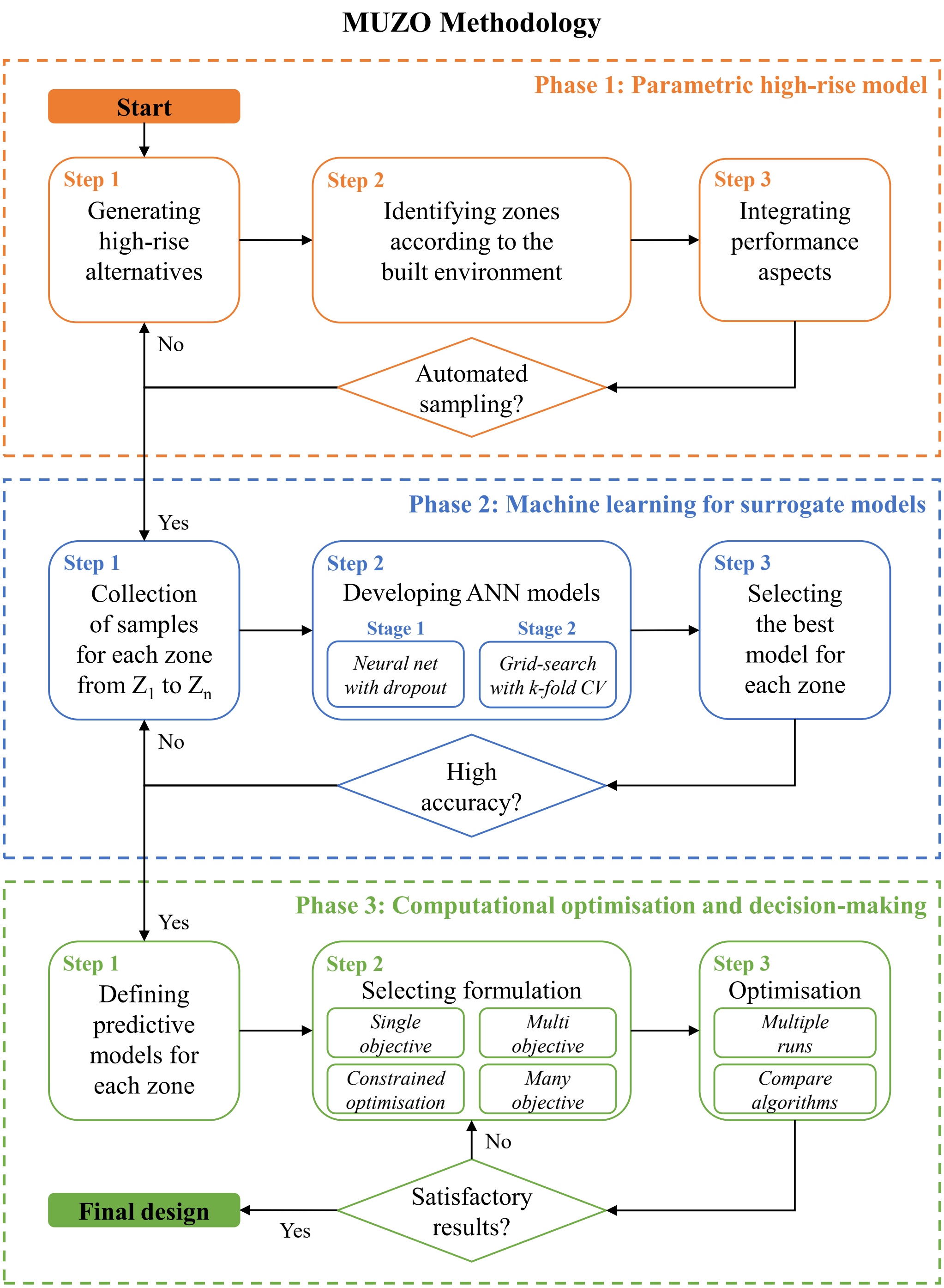


FIG. 4.1 MUZO methodology

TOC

Step 2 (Identifying zones)

As mentioned previously, dense urban districts result in performance variances between the ground and sky levels. Therefore, the second step of parametric modelling identifies the zones, which involves subdividing the entire building into smaller pieces to focus on various floor levels as different optimisation problems. The number of zones is a predefined variable that depends on the density of the plot under study. For instance, in urban areas with low-density, high-rise buildings can be divided into three zones, whereas this amount may increase to five in the mid-density scenarios. For high-density scenarios, more than five zones can be used for an extensive investigation of the effects of the surrounding at various levels. After determining the number of zones, the next step is to identify floor levels in each zone because performance aspects, such as daylight and solar radiation, require floor surfaces for the simulation to be conducted. While a large number of selected floor levels requires extensive simulation time, fewer selected floor levels may result in decision-making with limited awareness of the entire building's performance. Fig. 4.2 shows zoning scenarios for low-density, mid-density, high-density urban areas and different selections of floor levels.

Step 3 (Integrating performance aspects)

The final step of the first phase in MUZO methodology involves evaluating the high-rise model using the performance aspects of sustainable buildings. State of the art considers a limited number of performance criteria because of two reasons. First, considering multiple aspects requires extensive computational time for simulation-based evaluation. Second, the complexity of the design task increases owing to multiple performance aspects. In addition, conflicting performances introduce an additional challenge during the conceptual phase [35]. The proposed MUZO methodology can integrate any performance criteria to determine sustainable high-rise alternatives. Challenges on computational burden and complexity are addressed in the subsequent phases.

4.3.2 Machine learning for surrogate models

When the parametric model and simulations are set, various design alternatives can present the simulation results to gain awareness of the performance for different design scenarios. ANN models, which can swiftly evaluate the building performance, are used in the second phase of MUZO, which requires three steps:

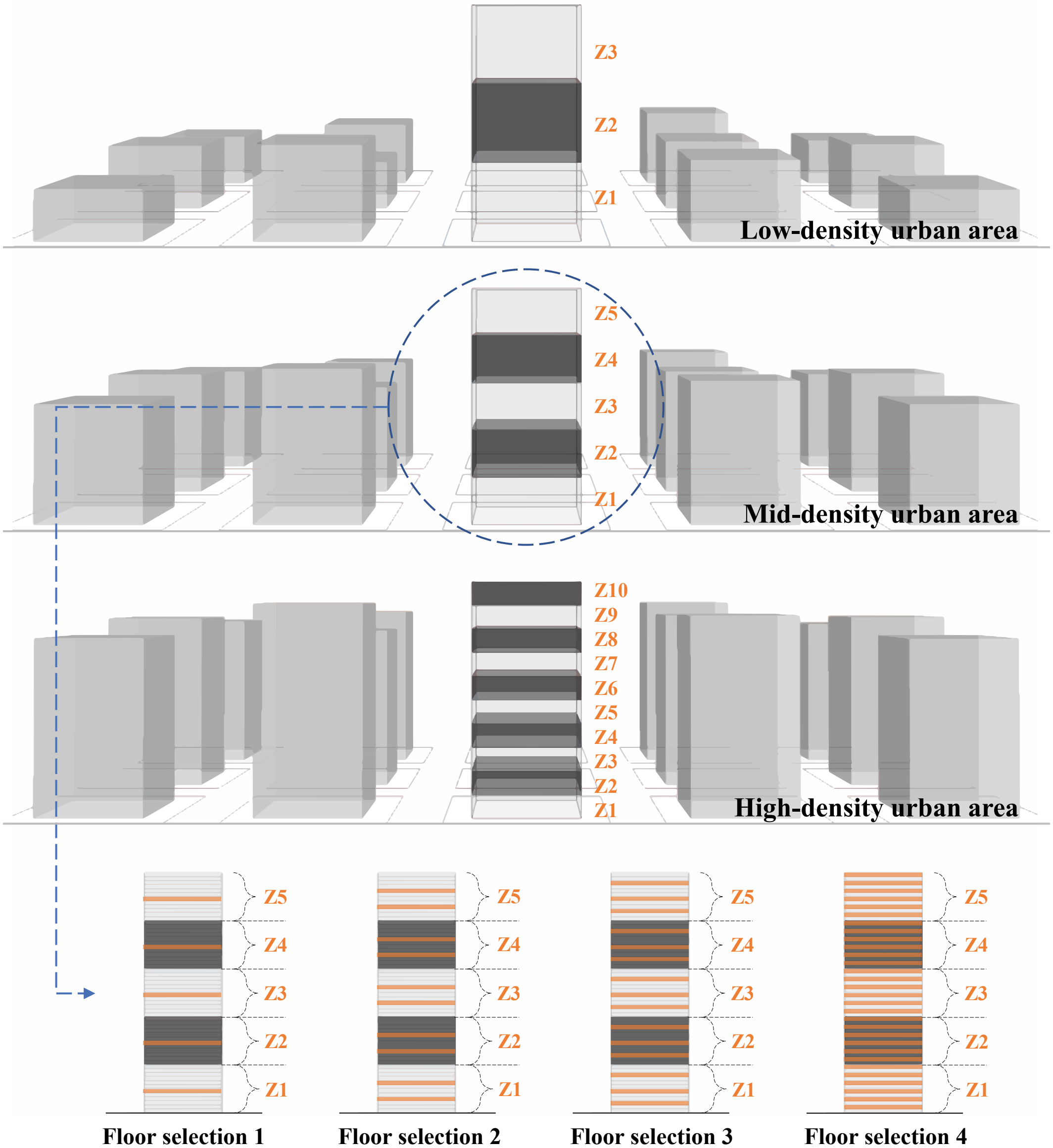


FIG. 4.2 Zoning examples for various urban densities and floor selections

Step 1 (Collection of samples)

Sampling, which is the first step of ML in MUZO, is an essential process of surrogate modelling. With a specific distribution in the data, ML algorithms can learn and predict data with high accuracy. Recently, Westermann and Evins [12] presented two types of sampling: static sampling (e.g. Latin hypercube sampling [36]) and adaptive sampling (i.e. sequential space-filling [37]). The selection of the sampling method depends on the category of surrogate models that can be either global or local models. All sampling methods using a global modelling approach can be used in the MUZO methodology. On the effect of the sample size, Chatzikonstantinou and Sariyildiz [38] discussed that the extension of the dataset is frequently beneficial. In addition, Roman, et al. [39] presented the most commonly used sampling methods in building-performance simulations. Among these sampling methods, one of the most common approaches is

$$n_s = 22.5 n_i \,, \tag{4.1}$$

where n_s is the sample size and n_i is the number of independent variables. Because each subdivided part corresponds to a different optimisation model, the MUZO methodology proposes a unique sample collection framework (Fig. 4.3). After subdividing the high-rise building into zones, each subdivision is used to generate its own samples. When the process is complete, each generated sampling file, which belongs to one zone, can be used in different surrogate models.

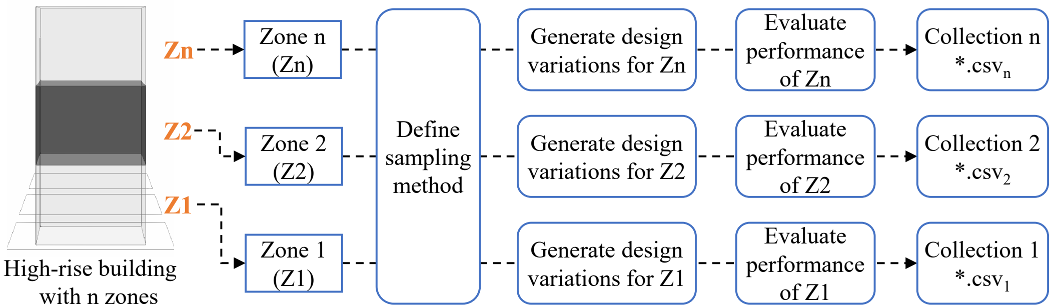


FIG. 4.3 Sample collection framework

TOC

Step 2 (Developing ANN models)

ANNs, which correspond to the second step of the ML phase, are widely used methods in ML domains to predict various aspects of building performance. This is because ANNs can manage large sample sizes for many variables and predict the performance with high accuracies [12]. Various ANN types, such as feedforward neural networks (FNNs) and radial basis function neural networks (RBFNNs), have been used to estimate building performance [39]. In this paper, the development of ANN models consists of two stages:

Stage 1 (Neural net with dropout): The development of ANNs begins with reading and scaling the data, which frequently contain different parameters with units and metrics. After the reading process, scaling is performed to obtain all inputs and outputs within the same boundaries using several scaling methods [40]. For min–max scaling, the data is normalised as

$$x' = \sigma\left(\max(x) - \min(x)\right) + \min(x), \tag{4.2}$$

where x' is the scaled value, x is the original value and σ is its standard deviation. Before selecting the scaling method, the problem type must be identified, which can be either classification or regression. While classification problems focus on predicting a class label, regression problems consider predicting a quantity. In addition, splitting data is crucial for identifying training and test sets using a rate, e.g. 0.2. or 0.25, according to Westermann and Evins [12]. When the ANN model is finalised, the architecture contains various layers (Fig. 4.4).

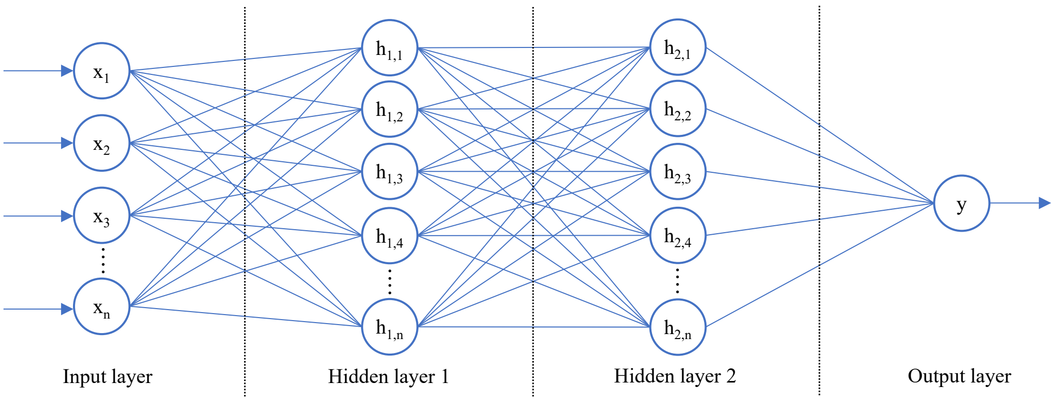


FIG. 4.4 ANN architecture

Each neurone in the hidden layers receives the weighted sum of inputs to pass the result through an activation function. In the output layer, the ANN predicts the final solution considering this procedure for all neurones. The network also involves bias layers that can shift the result of each layer. The activation (a) of each ith layer is

$$a_i = f\left(b_i + \sum_{j=1}^{m} w_{ij} x_i\right), \tag{4.3}$$

where f is the activation function, b is the bias, w_{ij} is the ith layer of the jth weight, and x_i is the input vector of the ith layer. When using rectified linear units (ReLU), each neurone is activated as follows:

$$f(x) = \begin{cases} 0 & for\ \ x \leq 0 \\ x & for\ \ x > 0 \end{cases}. \tag{4.4}$$

Different functions, e.g. sigmoid, softplus, and tanh, activate the neurones with various equations that may affect learning performance. The forward process of ANNs can predict the solution using Equation (4.3). To achieve high accuracy, a backward process is necessary to determine the best values for the weights and biases. Hence, backpropagation [41] involves a loss function and an optimisation algorithm. Researchers have widely used gradient descent (GD) [42], stochastic gradient descent (SGD) [43], Adam [44], and RMSProp [45] algorithms for optimisation. For the loss functions of classification problems, cross-entropy [46] and Kullback–Leibler divergence [47] can be considered. In regression problems, researchers use the mean squared error (MSE) in Equation (4.5) [48], mean absolute error (MAE) in Equation (4.6) [49], and R-squared (R^2) value in Equation (4.7) [50]:

$$MSE = \frac{1}{n}\sum_{i=1}^{n}(x_i - y_i)^2 \ , \tag{4.5}$$

$$MAE = \frac{\sum_{i=1}^{n} | y_i - x_i |}{n}, \tag{4.6}$$

$$R^2 = 1 - \frac{\sum_{i}(x_i - y_i)^2}{\sum_{i}(x_i - \bar{x})^2}, \tag{4.7}$$

where y_i is the predicted data, x_i is the observed data, $\bar{x}$ is the mean of the observed data, and n is the sample size. Various loss functions can be used to validate the accuracy of the trained model [38]. In addition, the dropout technique [51], which randomly drops units from the neural network with their connections, avoids overfitting. The MUZO methodology can involve multiple loss functions and dropouts at a rate between 0 and 1.

Stage 2 (Grid search with k-fold cross-validation (CV)): Developing a surrogate model is a black-box process. One of the reasons is that multiple hyperparameters, which are the parameters of the ANN (e.g. neurone size and batch size), are involved in the learning process. Various combinations of these factors affect learning and prediction accuracies. Therefore, a model validation technique is required to evaluate the accuracy of predictions. K-fold CV [52] is a well-known method for accurate estimations that randomly divides the original sample into k equal-sized subsamples. While one subsample is maintained as the test set, the remaining $k-1$ subsamples are the training sets. The standard deviation (Std) indicates the difference for each error for the k-fold CV:

$$Std = \sqrt{\frac{1}{N-1}\sum_{i=1}^{N}(x_i - \bar{x})^2}\ , \tag{4.8}$$

where N is the number of observations, $\{x_1,...,x_N\}$ are the observed values, and $\bar{x}$ is the mean value of these observations. The aim is to determine satisfactory results for the MAE, MSE, and R^2, and achieve small Std values for each accuracy metric.

Step 3 (Selecting the best model)

The final step of the ML phase involves the selection of the best model using the results of the grid search. In each zone, the criteria are the highest R^2 with low MAE, MSE, and Std for this process. Using the weights and biases of the final ANN, the predictive models are ready for use in the optimisation process.

4.3.3 Computational optimisation and decision-making

The final phase of the MUZO methodology, which consists of three steps, involves determining the design parameters for sustainable high-rise alternatives. The first step considers the development of predictive models using ML outputs. The second step is selecting the problem formulation. Finally, the proposed decision-making framework reveals the optimised design solution by completing the MUZO methodology.

Step 1 (Defining predictive models)

The development of predictive models requires weight and bias results collected from each ANN model. Subsequently, the collected results are transformed into matrices considering the input vector and neurone sizes for each layer to initiate the first step of the optimisation phase. The definition of activation with n layers is as follows.

$$y = f_n(f_{n-1}(f_{n-2}(f_{n-3}(x \cdot w_{n-3} + b_{n-3}) \cdot w_{n-2} + b_{n-2}) \cdot w_{n-1} + b_{n-1}) \cdot w_n + b_n), \quad (4.9)$$

where y is the performance criterion to be predicted, x is the input vector, w_n is the nth weight, b_n is the nth bias, and f_n is the nth activation function. For any given x, the model estimates the performance results. Having weights and biases as recorded data suggests the possibility of using predictive models in various platforms, such as C#, C++, Python, and GH, during the optimisation process.

Step 2 (Selecting formulation)

When the predictive models are ready, the next step is selecting the problem formulation for the optimisation process. Previous studies on building optimisation used single objective, weighted summation, multi-objective, many objectives, and constrained optimisation problems [53]. For n parameters, the definition of the generalised problem formulation is

$$\begin{array}{lll} \min: & f_1(X),\ldots,f_k(X), & X=(x_1,x_2,\ldots,x_n) \;\; and \;\; X \in S \\ subject\ to: & g_i(x) \le 0, & i=1,\ldots,p \\ & h_j(X)=0, & j=p+1,\ldots,m \end{array}, \quad (4.10)$$

where an integer $k > 0$ is the number of objective functions, S is the entire search space, p is the number of inequality constraints and $m-p$ is the number of equality constraints. For the maximisation problem, the transformation of the function can be

$$\max f(x) = -f(x). \quad (4.11)$$

The trade-off between building performance affects the selection of the formulation. For one aspect, e.g. maximising daylight [54], the single-objective formulation is convenient for the optimisation process. Another scenario may have two conflicting objectives, such as maximising the sDA and minimising the ASE. If one of these aspects requires a threshold according to the building standards, the formulation can be a single-objective constrained optimisation [55]. Otherwise, the multi-objective [56] or weighted summation [57] approaches can be alternatives. For more than three objectives, the options are multi-objective constrained or many-objective formulations [58].

Step 3 (Optimisation)

The final step of phase 3 involves exploring the optimal alternative for each zone. In the optimisation domain, heuristic algorithms are employed to solve complex problems by mimicking behavioural patterns and social phenomena observed in nature [59]. Additionally, in the domain of sustainable building design, heuristics are widely used because promising alternatives are discovered in a reasonable time frame [9]. Despite their advantages, these algorithms do not guarantee an optimal solution. According to the no free lunch (NFL) theorem [60], a global algorithm that can determine the optimal result for all problems does not exist. In architectural design, the subject is more dynamic than the benchmark problems. Each design scenario is a specific problem owing to the variances in the surroundings. In addition, the surroundings of the different cities have diverse climate types (e.g. Mediterranean climate in Izmir, Oceanic climate in Amsterdam). Therefore, architects can propose various alternatives for the same design problem (i.e. high-rise buildings) because concerns and the required strategies are different. Thus, we may conclude that "*the global optimal of each design problem is unexplored*". Therefore, the optimisation process of the MUZO methodology involves comparing various algorithms with replications for decision-making (Fig. 4.5). Single-objective optimisation algorithms report the best solution that can be used as the final design alternative. In multi-objective or many-objective optimisation problems, various post-optimisation analysis methods can be used to evaluate the quality of the single best solution during the decision-making process [61], e.g. weighted summation approach [62], TOPSIS [63], analytic hierarchy process [64], minimum distance to the utopic point [65], auto-associative models [66].

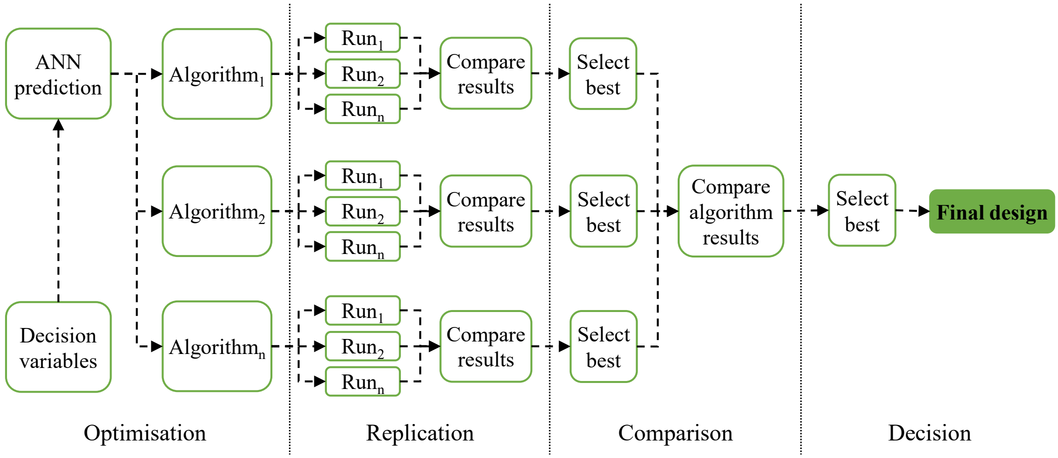


FIG. 4.5 MUZO optimisation process and decision framework

4.4 Setup of the case study

This section explains the setup for evaluating the MUZO methodology considering a hypothetical dense urban district. The first subsection describes a parametric high-rise building with variables for the two façade types. The subsequent subsection presents the selected performance aspects of the simulation setup. Finally, surrogate modelling introduces the details of the sample collection and the development of ANN models.

4.4.1 Parametric high-rise model and the built environment

The hypothetical district had 25 plots in GH, each with a 2500 m^2 footprint with building heights between 50 and 150 m, which were generated randomly. The focus of the study was the central plot with 60 floors, 2100 m^2 net one-floor area, 150,000 m^2 gross floor area, and 50 × 50 m façade length. Fig. 4.6 shows the subdivisions (zones) of the building beginning from ground-level zone 1 (Z1) to sky-level zone 10 (Z10), as well as selected floor levels (second and fifth) of every zone for simulations. Table 4.1 presents the façade, shape, and glazing parameters used in the parametric models for both scenarios.

TOC

The first façade design focussed on horizontal and vertical shading devices using the number, length, and rotation of the devices with four glazing types. The second design considered diagonal shading devices involving the number, length, and rotation of first and second-order diagonals with the same glazing types. The design setups in Figs. 4.7 and 4.8 were used for each orientation, i.e. north (N), south (S), east (E), and west (W). Including floor-to-floor height and rotation parameters, the search space for the quad-grid scenario in one zone had 2.893399115e+28 design alternatives with 26 parameters, whereas this number was 3.054543465e+23 for the diagrid scenario with 22 parameters. Floor-to-floor height and rotation parameters of the lower zones affected the height and rotation of the higher zones. Therefore, the total amount of the design parameters in one zone increased from Z1 to Z10 (Fig. 4.9). Consequently, the quad-grid design had 26 parameters in Z1 and 44 variables in Z10, while the diagrid design had 22 parameters in Z1 and 40 variables in Z10.

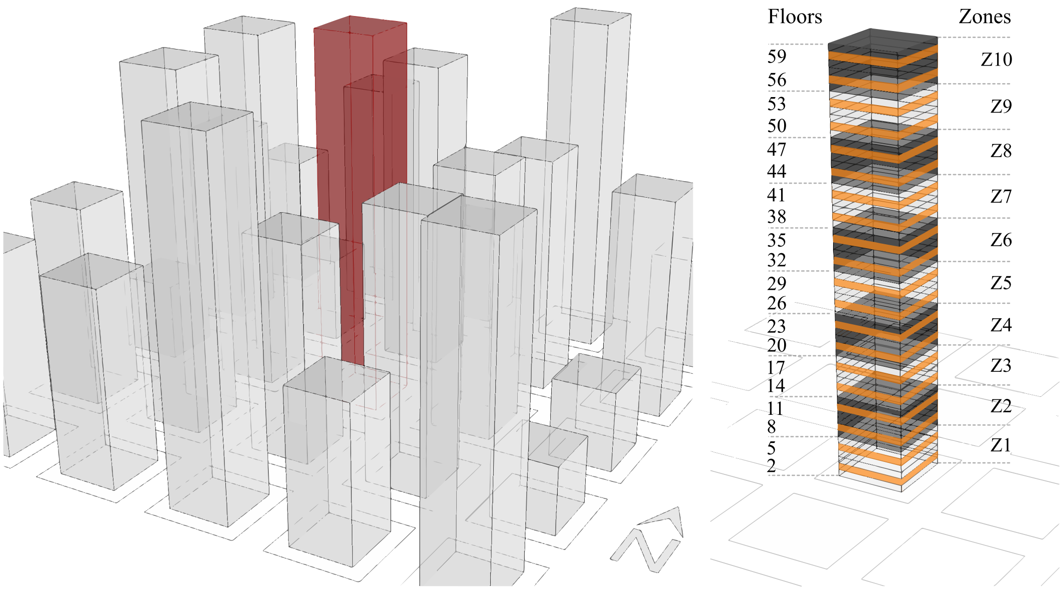


FIG. 4.6 Plot under study, zones (subdivisions) and selected floors of the high-rise for simulation

TOC

TABLE 4.1 Parameters of the quad-grid and diagrid scenarios

	Parameters	Explanation	Location	Type	Unit	Boundary
Quad-grid façade	x_{Q1}, x_{Q6}, x_{Q11}, x_{Q16}	Number of vertical devices	N-S-E-W	Discrete	-	[0,8]
	x_{Q2}, x_{Q7}, x_{Q12}, x_{Q17}	Length of vertical devices		Continues	m	[0.0, 1.5]
	x_{Q3}, x_{Q8}, x_{Q13}, x_{Q18}	Rotation of vertical devices		Discrete	°	[-60, 60]
	x_{Q4}, x_{Q9}, x_{Q14}, x_{Q19}	Number of horizontal devices		Discrete	-	[0,2]
	x_{Q5}, x_{Q10}, x_{Q15}, x_{Q20}	Length of horizontal devices		Continues	m	[0.0, 1.5]
	x_{Q21}, x_{Q22}, x_{Q23}, x_{Q24}	Glazing type		Discrete	-	[1,4]
Diagrid façade	x_{D1}, x_{D5}, x_{D9}, x_{D13}	Length of 1st order diagonal	N-S-E-W	Continues	m	[0.0, 1.5]
	x_{D2}, x_{D6}, x_{D10}, x_{D14}	Length of 2nd order diagonal		Continues	m	[0.0, 1.5]
	x_{D3}, x_{D7}, x_{D11}, x_{D15}	Rotation of diagonal devices		Discrete	°	[-60, 60]
	x_{D4}, x_{D8}, x_{D12}, x_{D16}	Number of diagonal devices		Discrete	-	[0,5]
	x_{D17}, x_{D18}, x_{D19}, x_{D20}	Glazing type		Discrete	-	[1,4]
Building shape	$x_1,\ldots,x_{10}$	Floor-to-floor height of zones	-	Continues	m	[4.0, 5.0]
	$x_{11},\ldots,x_{20}$	Rotation of zones		Discrete	°	[-10, 10]

	Type	Explanation	Tvis	U-val.	g-val.
Glazing types	1	Tinted float 8 mm blue - 12 mm air - Temperable Low-E 8 mm blue	0.22	1.6	0.28
	2	Temperable Low-E 8 mm neutral - 12 mm air - Clear float 8 mm - 12 mm air - Temperable Low-E 8 mm green	0.45	0.9	0.40
	3	Tinted float 8 mm green	0.68	5.6	0.51
	4	Ultra-clear float 8 mm - 12 mm air - Ultra-clear float 8 mm	0.82	2.8	0.81

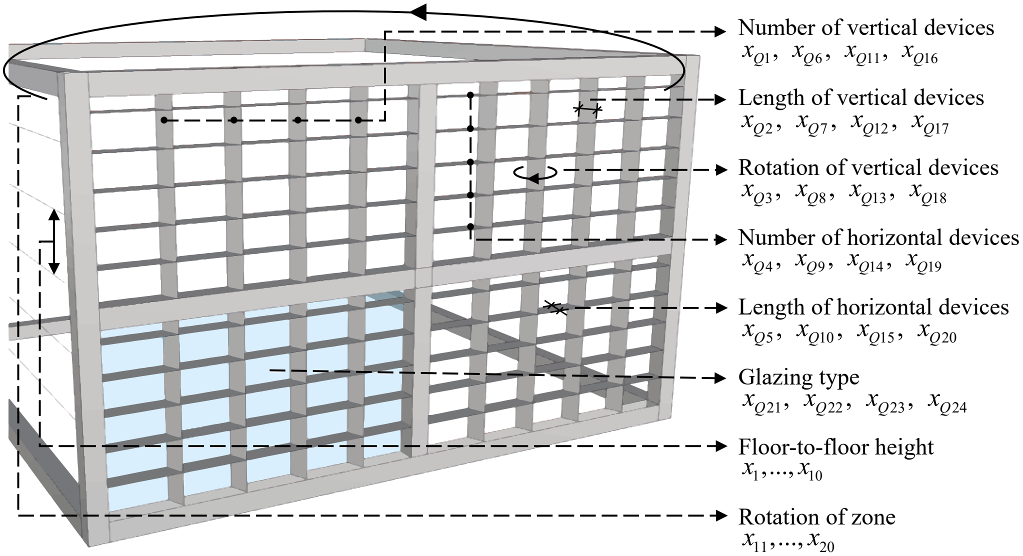


FIG. 4.7 Quad-grid façade design

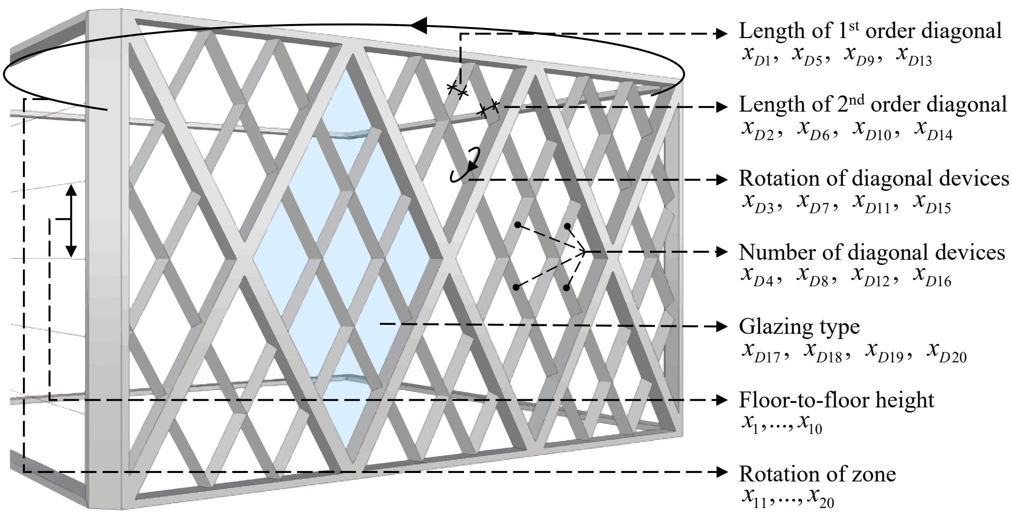


FIG. 4.8 Diagrid façade design

Zone -Case	Floor-to-floor height parameters										Rotation parameters										Façade parameters		Number of parameters	
	x_1	x_2	x_3	x_4	x_5	x_6	x_7	x_8	x_9	x_{10}	x_{11}	x_{12}	x_{13}	x_{14}	x_{15}	x_{16}	x_{17}	x_{18}	x_{19}	x_{20}	$x_{Q1,\ldots,}\,x_{Q24}$	$x_{D1,\ldots,}\,x_{D20}$	(Q)	(D)
Z1-Q	•										•										•		26	
Z1-D	•										•											•		22
Z2-Q	•	•									•	•									•		28	
Z2-D	•	•									•	•										•		24
Z3-Q	•	•	•								•	•	•								•		30	
Z3-D	•	•	•								•	•	•									•		26
Z4-Q	•	•	•	•							•	•	•	•							•		32	
Z4-D	•	•	•	•							•	•	•	•								•		28
Z5-Q	•	•	•	•	•						•	•	•	•	•						•		34	
Z5-D	•	•	•	•	•						•	•	•	•	•							•		30
Z6-Q	•	•	•	•	•	•					•	•	•	•	•	•					•		36	
Z6-D	•	•	•	•	•	•					•	•	•	•	•	•						•		32
Z7-Q	•	•	•	•	•	•	•				•	•	•	•	•	•	•				•		38	
Z7-D	•	•	•	•	•	•	•				•	•	•	•	•	•	•					•		34
Z8-Q	•	•	•	•	•	•	•	•			•	•	•	•	•	•	•	•			•		40	
Z8-D	•	•	•	•	•	•	•	•			•	•	•	•	•	•	•	•				•		36
Z9-Q	•	•	•	•	•	•	•	•	•		•	•	•	•	•	•	•	•	•		•		42	
Z9-D	•	•	•	•	•	•	•	•	•		•	•	•	•	•	•	•	•	•			•		38
Z10-Q	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•		44	
Z10-D	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•		•		40

● (Q) Quad-grid parameters ● (D) Diagrid parameters

FIG. 4.9 Complexity matrix for quad-grid and diagrid scenarios

4.4.2 Performance metrics and simulation setup

We investigated two of the LEED v4.1 metrics for the case buildings, namely, the sDA and ASE, introduced for the green building certification program [67]. Both metrics are commonly used for various building functions to achieve sustainable design solutions [68-73]. Recently, Illuminating Engineering Society (IES) presented definitions for sDA and ASE metrics [74]. The sDA evaluates the annual efficiency of ambient daylight levels in interior spaces. The calculation method results in the percentage of an analysis area with a minimum daylight illuminance level for specific hours. In contrast, the ASE indicates the potential visual discomfort in interior work environments. The method results in the percentage of direct sunlight that exceeds a defined illuminance for the specified number of hours for the analysis area.

TOC

The simulation setup focussed on the second and fifth floors of each zone (Fig. 4.10). The parametric model used the Diva plug-in v4.0.3.1 [75] developed for GH to simulate sDA and ASE metrics with an EnergyPlus weather data file for Izmir City with a dry summer Mediterranean climate, latitude: 38.423733 and longitude 27.142826. Each zone was simulated using two analyses of planes with 180 sensor points each and was 0.8 m above the finished floor. Four glazing types, listed in Table 4.1, were separately used as decision variables for each orientation. As suggested by [74], the setup simulated $sDA_{300/50\%}$ and $ASE_{1000,250h}$ for 10 h of occupation between 8 am and 6 pm. A single simulation task of one zone involved 360 sensors. For the radiance parameters listed in Table 4.2, the simulation process used values similar to those in previous studies because of the high computational cost. The setup was used to simulate the daylight performance of 7200 sensor points for the overall building evaluation of two scenarios.

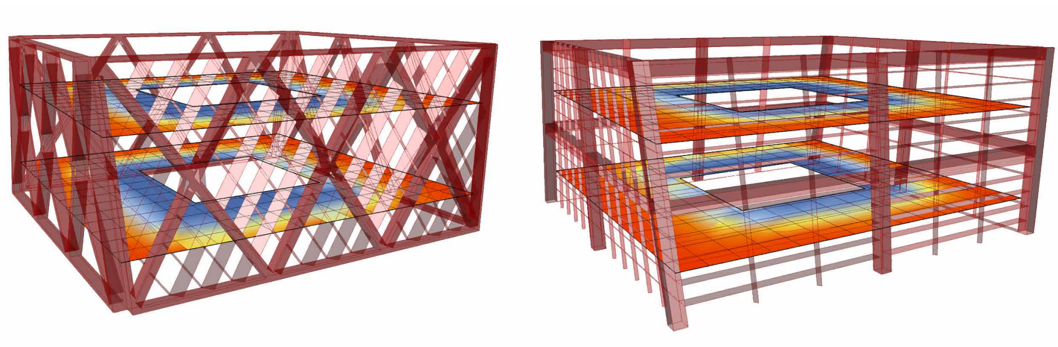


FIG. 4.10 Simulation setup for both scenarios

TABLE 4.2 Radiance parameters

	-aa	-ab	-ad	-ar	-as
This paper	0.15	2	512	256	128
[76]	0.15	2	512	256	128
[77]	0.15	2	1000	300	20
[56]	0.15	2	512	256	128
[78]	0.15	2	512	256	128

4.4.3 Surrogate modelling

The surrogate modelling began with sampling collection, which considered 1000 samples for each zone using Latin hypercube sampling [36] and Equation (4.1). One simulation required 4 min for two floors with the radiance parameters provided in Table 4.2. A computer with an Intel I7 4 core processor at 2.7 GHz and 16 GB DDR3 memory was used to calculate the computational burden as more than 55 days were required to collect 20,000 samples. In the next step, Python 3 [79] was used with the additional libraries listed in Table 4.3 to develop ANN models with FNNs. After scaling the data for each zone with min-max scaling in Equation (4.2), the SGD algorithm optimised weights and biases using Equation (4.3) for all models considering 10-fold CV, three hidden layers, dropout rate with 0.1, and the ReLU activation function in Equation (4.4). The automated Python program fit the model 324 times for all hyperparameter combinations for every zone. In total, the program ran 6,480 different ANN models with various complexities.

TABLE 4.3 Python libraries

Library name	Explanation	Reference
Scikit-learn	ML library	[80]
Keras	Deep learning library	[81]
TensorFlow	Open source ML platform	[82]
Pandas	Data analysis library	[83]
Joypy	Plot library	[84]
Plotly	The interactive graphing library	[85]
Matplotlib	Static, animated, and interactive visualisation library	[86]

4.5 Results

This section presents the sampling, grid search with CV, and tuned ANN results. The supplementary material provides statistics of collected samples, selected ANN models with learning scores, weights, biases, and computation time spent on the model selection for each zone.

4.5.1 Sampling results

The collected samples, which were published as an open-access dataset in [87], contain the simulation results for the quad-grid and diagrid. Each zone had ASE and sDA results indicated as ASE_1 and sDA_1 on the second floor and ASE_2 and sDA_2 on the fifth floor. ASE_avg and sDA_avg, which represent the average values of these floors (Fig. 4.11), were used to develop the surrogate models.

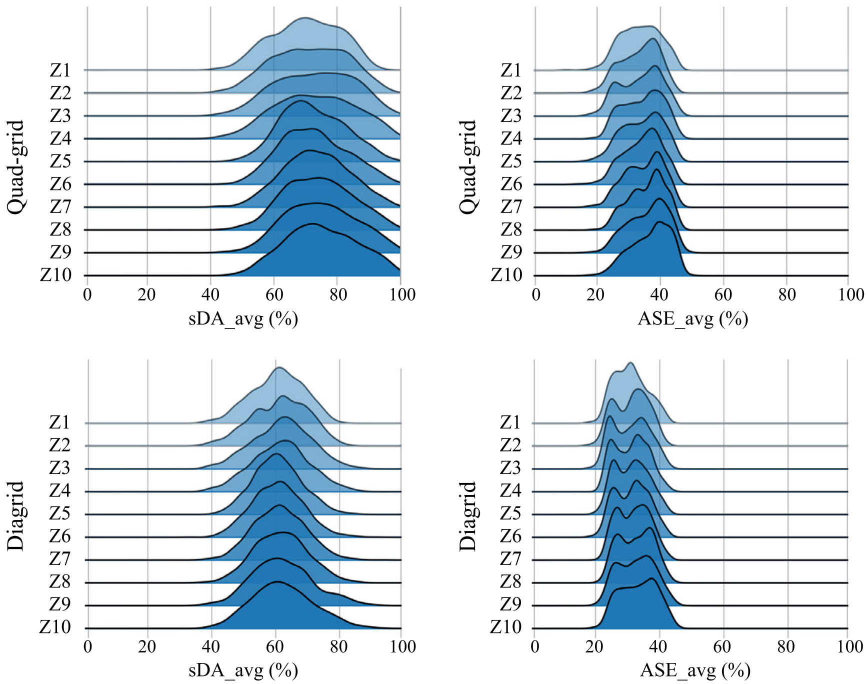


FIG. 4.11 Distributions of collected samples

TOC

The sDA results of the quad-grid application were between 41.9% and 100%, whereas ASE results were in the range of 9.4% and 50.7%. In the diagrid scenario, these results were 33.1% and 93.75% for sDA and 16.75% and 46.2% for ASE. For the mean values, natural daylight availability increased from Z1 to Z10 for both scenarios. However, this caused an increment in ASE results. In addition, the means of the sDA results for the quad-grid were higher than those for the diagrid.

4.5.2 Grid search with cross-validation results

ANN models were trained using the developed Python program, considering grid search and 10-fold CV using the collected dataset. The average results and deviations of MSE, MAE, and R^2 were recorded for each parameter combination during the search process. The best hyperparameters with their statistical results are shown in Fig. 4.12.

The results indicated that 37 out of 40 ANN models had the best accuracy using 200 neurones. In the three models, the number of neurones was 100. For the momentum parameter, 33 models had the best score with 0.9, five models had 0.6, and two models had 0.3. Additionally, 25 models had the highest score using 0.1 for the learning rate, while twelve models using 0.05, and three models using 0.01. For epochs, eighteen models had the highest accuracy with 500, fourteen models had 750, six models using 1000, and two models had 250. Finally, the best selection for the batch size was 50 in twenty-one models, whereas it was 100 in nine models and 10 in ten models. The deviations of MAE, MSE, and R^2 indicated that all CV folds had similar results for all metrics with high accuracies. The R^2 values of 33 models were higher than 0.9, whereas in seven models they were higher than 0.8. All MAE, MSE, and Std results were less than 0.05. Consequently, the grid search results indicated promising accuracies to develop predictive models with the selected hyperparameters in the next step.

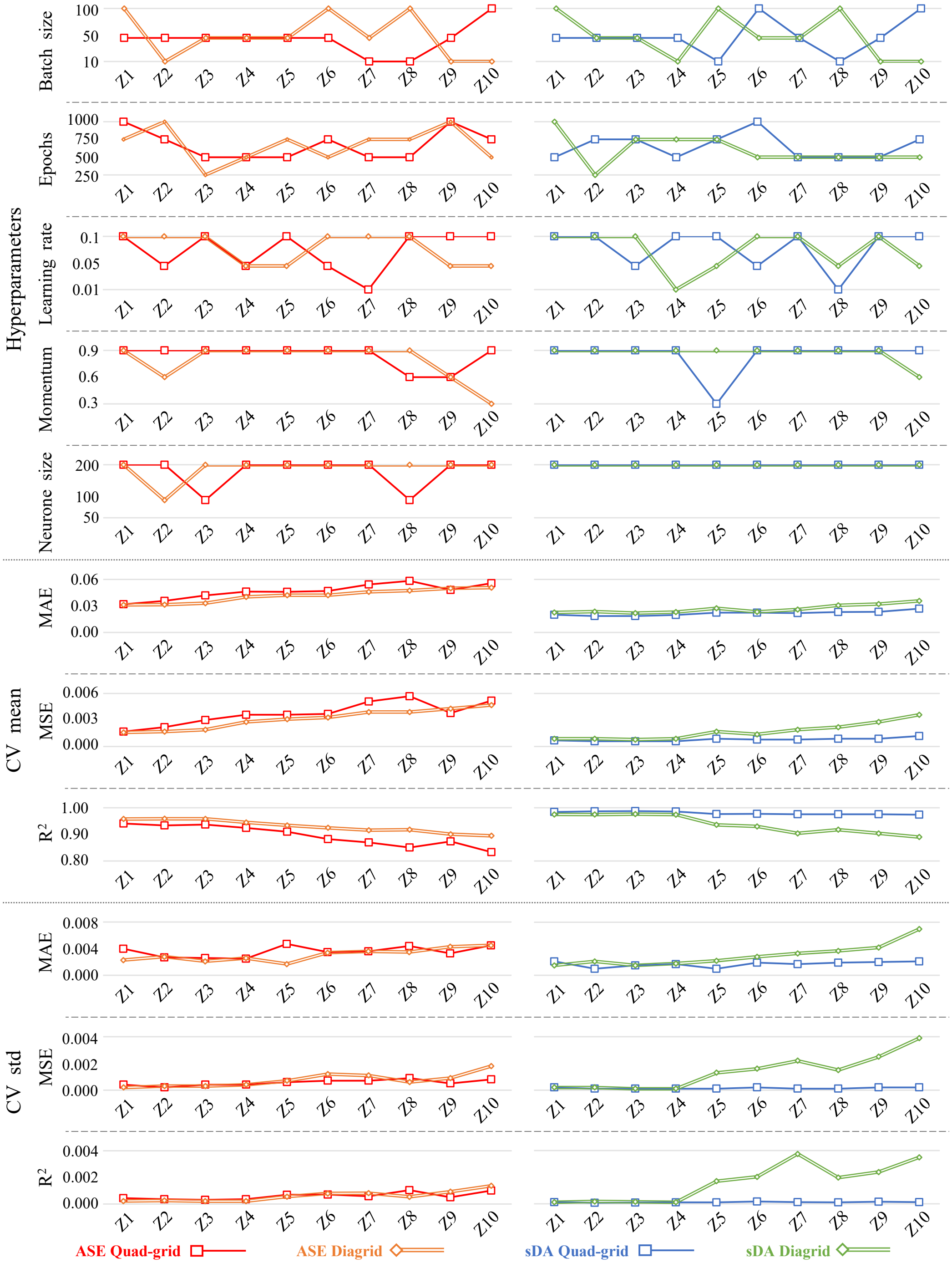


FIG. 4.12 Grid search results of the best ANN models

4.5.3 Tuned ANN results

Using the best hyperparameter sets shown in Fig. 4.12, ANN models were fit considering 0.2 for splitting data to demonstrate the learning behaviour and convergence by separating the dataset as training and test sets. Fig. 4.13 shows the R^2 results of these models, and appendices 4A and 4B provide the convergence of MSE and MAE while fitting the ANN models.

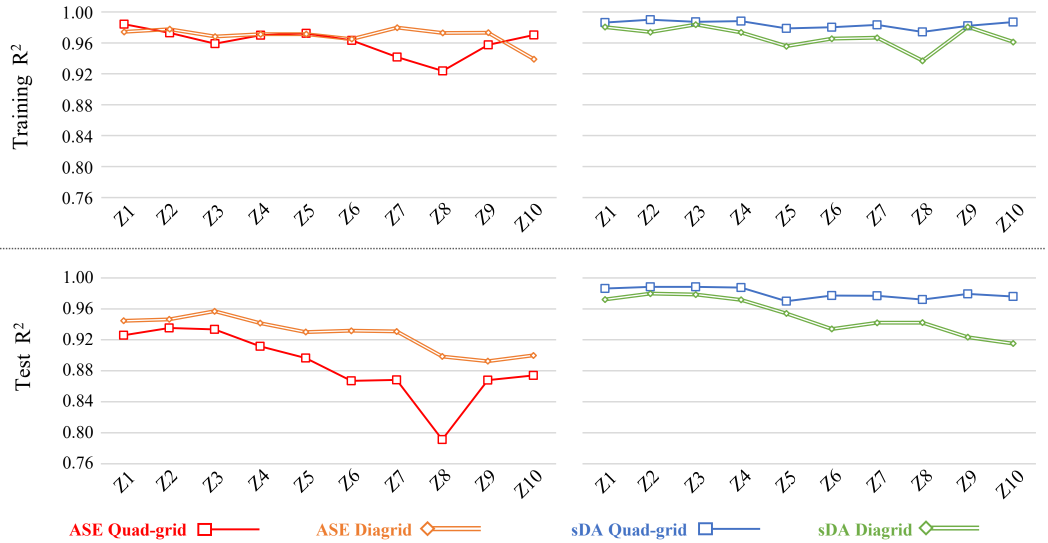


FIG. 4.13 R^2 results of training and test sets

The R^2 values for all training sets were higher than 0.9. For the test sets, R^2 of fourteen quad-grid models out of twenty were higher than 0.9 and higher than 0.8 for five models. R^2 was slightly lower than 0.8 for only one model. In the diagrid application, R^2 for seventeen models was higher than 0.9, while it was higher than 0.8 for three models. The accuracy of the predictive models through the MSE and MAE results are also provided in appendices 4A and 4B. All reported MSE results were lower than 0.05. For the MAE, results of the ASE and sDA were lower than 0.05 in Z1, Z2, Z3, Z4, Z6, Z9, and Z10 for the quad-grid scenario. However, in other zones, the MAE of the ASE was slightly larger than 0.05, and it was lower than 0.05 for the sDA. In the diagrid scenario, the ASE and sDA models had MAE results lower than 0.05 in Z1, Z2, Z3, Z4, Z5, Z6, Z7, and Z9. In other zones, the MAE of the ASE was slightly higher than 0.05, whereas it was smaller than 0.05 for the sDA.

TOC

To compare the accuracy results reported for different design complexities, Fig. 4.14 shows the R^2 of similar studies focusing on ML applications in daylight. Papers in this domain have promising results for DA, sDA, illumination level (IL), and useful daylight illuminance (UDI). However, visual comfort metrics, such as daylight glare probability (DGP), have moderate accuracies for various ML applications. In this study, a similar result was achieved for the ASE metric because of the challenges in predicting comfort metrics. In addition, most of the previous studies considered design variables between 5 and 15, which provided less design complexity compared with this study. Only Kirimtat, Krejcar, Ekici and Tasgetiren [77] considered 25 variables with R^2 values between 0.9 and 0.3. Consequently, the ML part of the MUZO methodology could address more complex designs while presenting high accuracies for all 40 models.

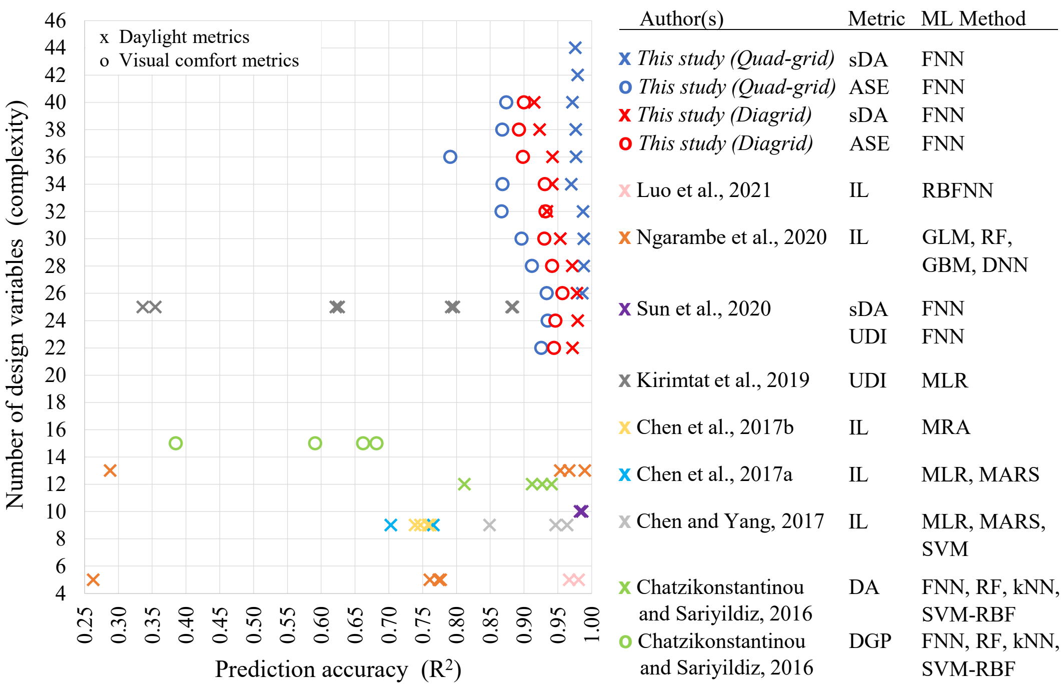


FIG. 4.14 Summary of the previous studies for ML applications in daylight

TOC

4.6 Conclusion

This paper presents the first part of the MUZO study, focusing on the background, methodology, setup, and ML results. The proposed methodology managed sampling and ANN development for 40 different models using a parametric high-rise model in a dense urban district. In addition, the developed Python program was used to investigate the best models for all zones of the two scenarios in 403 h. Based on the reported accuracies, building zones close to the sky levels were more challenging than the ground levels because of the increasing number of design variables. The study also proved that dense urban surroundings affect the performance of high-rise buildings at various floor levels by determining different simulation results during the sampling process. Therefore, architects and engineers should consider various zones as different problems while designing sustainable high-rises in metropolises.

The ML part of the MUZO methodology indicated prediction scores with high accuracies using different hyperparameters for batch size, epoch, neurone size, momentum, and learning rate in each model despite various design complexities considering multiple performance aspects. Future research can integrate more hyperparameters, such as activation function, dropout rate, various optimisation algorithms, different numbers of hidden layers, and sample sizes. Thus, the ANN models can provide higher accuracies with an exponential increment in computational time. Hence, having all these parameters can be more applicable to real-world high-rise scenarios.

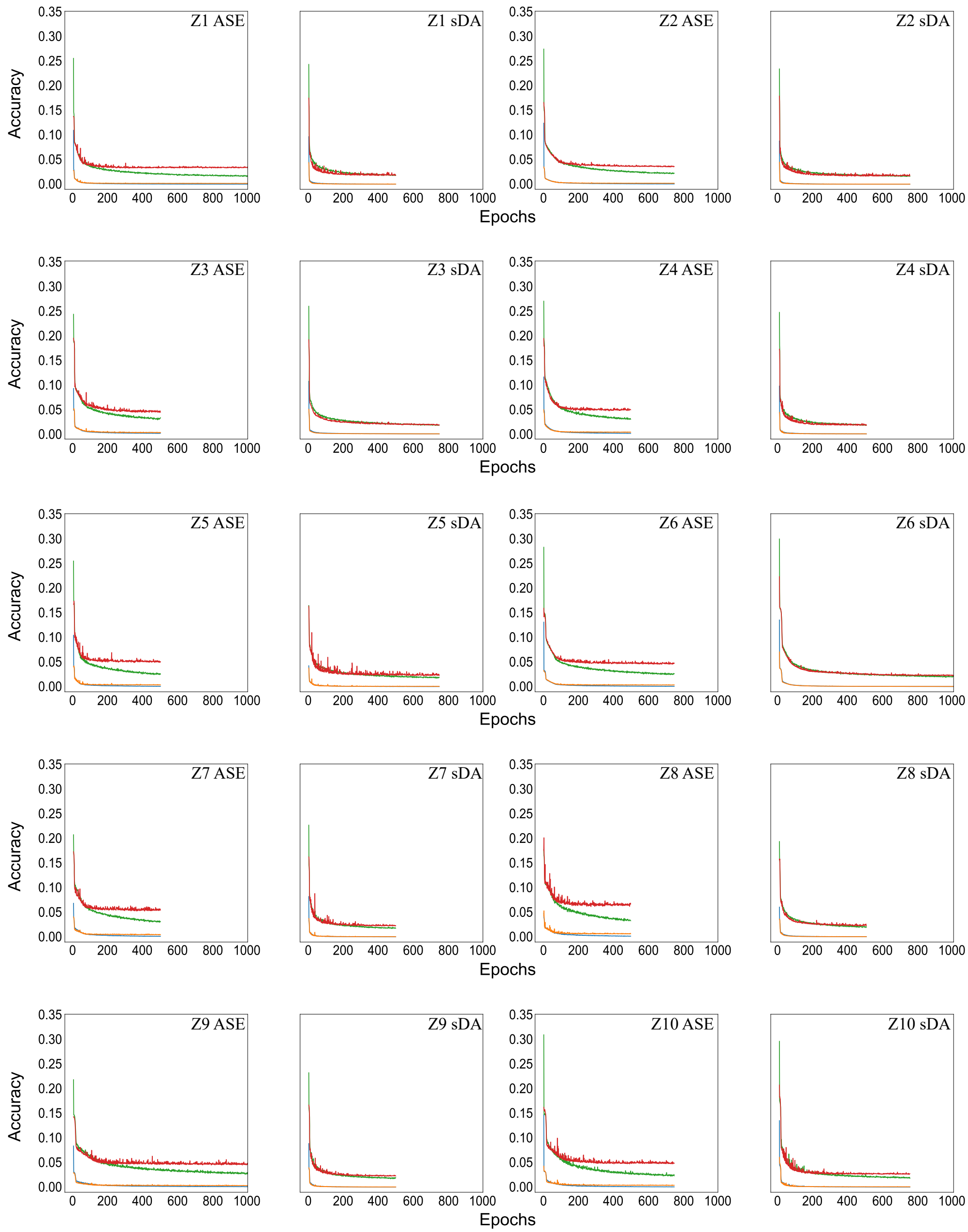
In conclusion, the parametric high-rise model and ML for surrogate model phases of the MUZO methodology could automate form generation, performance evaluation, sampling, data processing, ANN development, and reporting the predictive models for all zones in both high-rise scenarios. Using the ML part of the MUZO methodology, architects and engineers can address the computational burden while optimising the entirety of a high-rise building to propose sustainable alternatives in metropolises. Nevertheless, optimisation of the high-rise building, which is addressed in part 2 of this study, remains challenging owing to the high number of parameters involved in the design process.

Data availability

Datasets related to this article can be found at (an open-source online data repository hosted at 4TU Research Data [87]): https://doi.org/10.4121/uuid:8538ac2f-3a78-4923-8fca-5beb50017299

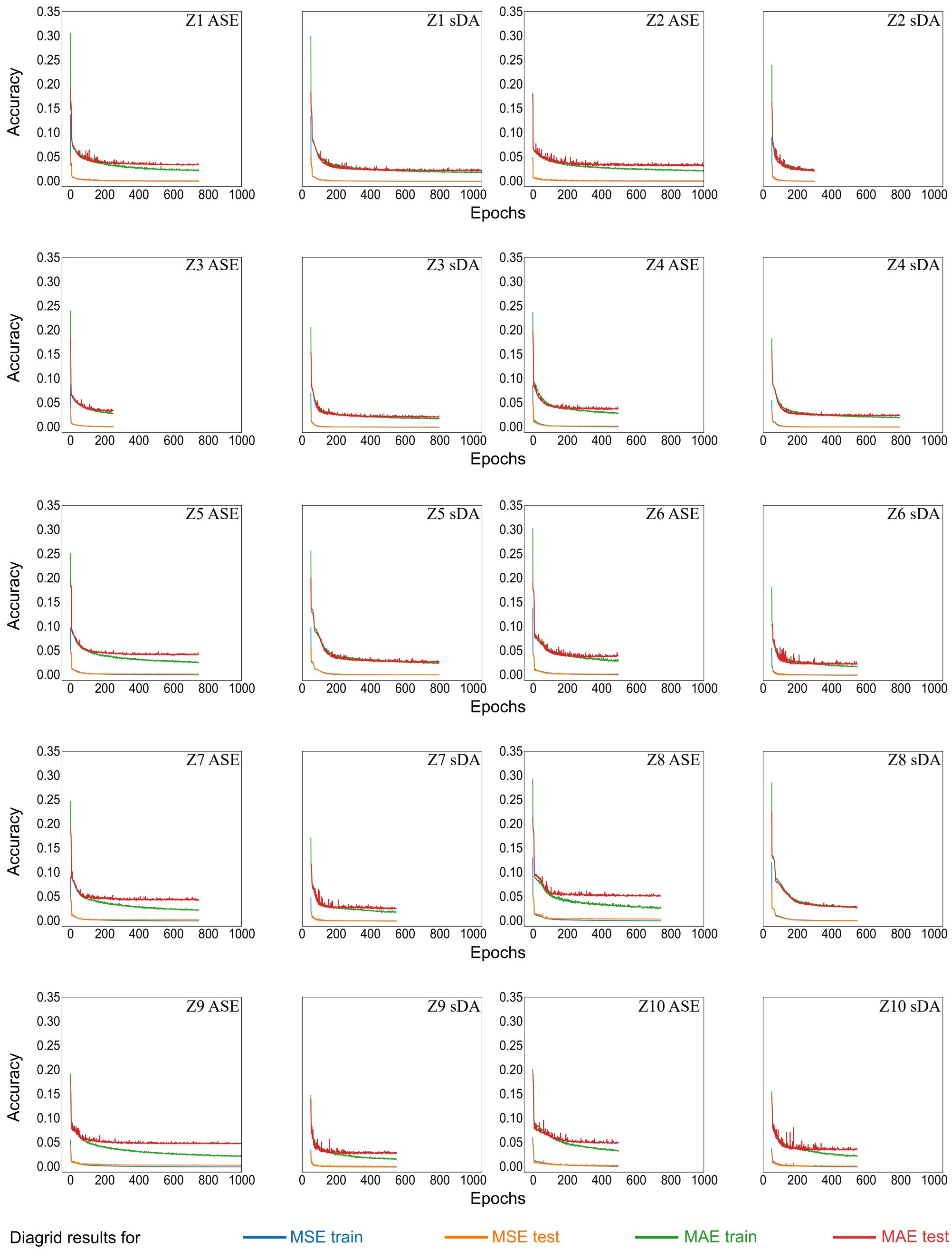
TOC

Appendix 4A. Quad-grid MAE and MSE results for training and test sets



Quad-grid results for MSE train MSE test MAE train MAE test

Appendix 4B. Diagrid MAE and MSE results for training and test sets



References

[1] Al-Kodmany, K.; Ali, M. M., *The future of the city: Tall buildings and urban design*. WIT press: **2013**.

[2] Cohen, B., Urbanization in developing countries: Current trends, future projections, and key challenges for sustainability. *Technology in society* **2006,** 28, (1-2), 63-80.

[3] UN, Department of Economic and Social Affairs, Population Division. World Urbanization Prospects: The 2018 Revision (ST/ESA/SER.A/420). New York: United Nations. **2019**.

[4] CTBUH, The Skyscraper Center, Council on Tall Buildings and Urban Habitat, http://www.skyscrapercenter.com (Last Seen: 03.07.2020). **2020**.

[5] IEA, Tracking Buildings 2020, IEA, Paris https://www.iea.org/reports/tracking-buildings-2020. **2020**.

[6] Godoy-Shimizu, D.; Steadman, P.; Hamilton, I.; Donn, M.; Evans, S.; Moreno, G.; Shayesteh, H., Energy use and height in office buildings. *Building Research & Information* **2018,** 46, (8), 845-863.

[7] Lee, M.; Brauer, M.; Wong, P.; Tang, R.; Tsui, T. H.; Choi, C.; Cheng, W.; Lai, P.-C.; Tian, L.; Thach, T.-Q., Land use regression modelling of air pollution in high density high rise cities: A case study in Hong Kong. *Science of the Total Environment* **2017,** 592, 306-315.

[8] Samuelson, H.; Claussnitzer, S.; Goyal, A.; Chen, Y.; Romo-Castillo, A., Parametric energy simulation in early design: High-rise residential buildings in urban contexts. *Building and Environment* **2016,** 101, 19-31.

[9] Evins, R., A review of computational optimisation methods applied to sustainable building design. *Renewable and sustainable energy reviews* **2013,** 22, 230-245.

[10] Sariyildiz, I. In *Performative computational design*, Keynote speech in: Proceedings of ICONARCH-I: International congress of architecture-I, Konya, Turkey, 15-17 November **2012**, Selcuk University.

[11] Asadi, S.; Geem, Z. W., Sustainable building design: A review on recent metaheuristic methods. *Recent Advances in Swarm Intelligence and Evolutionary Computation* **2015**, 203-223.

[12] Westermann, P.; Evins, R., Surrogate modelling for sustainable building design–A review. *Energy and Buildings* **2019,** 198, 170-186.

[13] Cubukcuoglu, C.; Ekici, B.; Tasgetiren, M. F.; Sariyildiz, S., OPTIMUS: Self-Adaptive Differential Evolution with Ensemble of Mutation Strategies for Grasshopper Algorithmic Modeling. *Algorithms* **2019,** 12, (7), 141.

[14] Wortmann, T., Opossum-introducing and evaluating a model-based optimization tool for grasshopper. **2017**.

[15] Lam, J. C.; Hui, S. C.; Chan, A. L., Regression analysis of high-rise fully air-conditioned office buildings. *Energy and Buildings* **1997,** 26, (2), 189-197.

[16] Ko, D. H.; Elnimeiri, M.; Clark, R. J., Assessment and prediction of daylight performance in high-rise office buildings. *The Structural Design of tall and special buildings* **2008,** 17, (5), 953-976.

[17] Li, Y.; Li, X. In *Natural ventilation potential of high-rise residential buildings in northern China using coupling thermal and airflow simulations*, Building Simulation; Springer: **2015**; pp 51-64.

[18] Tian, Z.; Wei, S.; Shi, X., Developing data-driven models for energy-efficient heating design in office buildings. *Journal of Building Engineering* **2020,** 32, 101778.

[19] Chen, X.; Yang, H.; Sun, K., Developing a meta-model for sensitivity analyses and prediction of building performance for passively designed high-rise residential buildings. *Applied energy* **2017,** 194, 422-439.

[20] Chen, X.; Yang, H.; Wang, Y., Parametric study of passive design strategies for high-rise residential buildings in hot and humid climates: miscellaneous impact factors. *Renewable and Sustainable Energy Reviews* **2017,** 69, 442-460.

[21] Imam, M.; Kolarevic, B., Towards Resource-Generative Skyscrapers. *Council on Tall Buildings and Urban Habitat research paper* **2016**.

[22] Chen, X.; Huang, J.; Yang, H.; Peng, J., Approaching low-energy high-rise building by integrating passive architectural design with photovoltaic application. *Journal of Cleaner Production* **2019,** 220, 313-330.

[23] Chen, X.; Yang, H.; Peng, J., Energy optimization of high-rise commercial buildings integrated with photovoltaic facades in urban context. *Energy* **2019,** 172, 1-17.

[24] Giouri, E. D.; Tenpierik, M.; Turrin, M., Zero energy potential of a high-rise office building in a Mediterranean climate: Using multi-objective optimization to understand the impact of design decisions towards zero-energy high-rise buildings. *Energy and Buildings* **2020,** 209, 109666.

[25] Liu, J.; Wang, M.; Peng, J.; Chen, X.; Cao, S.; Yang, H., Techno-economic design optimization of hybrid renewable energy applications for high-rise residential buildings. *Energy Conversion and Management* **2020,** 213, 112868.

[26] Gan, V. J.; Wong, H.; Tse, K. T.; Cheng, J. C.; Lo, I. M.; Chan, C. M., Simulation-based evolutionary optimization for energy-efficient layout plan design of high-rise residential buildings. *Journal of cleaner production* **2019,** 231, 1375-1388.
[27] Chen, X.; Yang, H., A multi-stage optimization of passively designed high-rise residential buildings in multiple building operation scenarios. *Applied energy* **2017,** 206, 541-557.
[28] Chen, X.; Yang, H., Integrated energy performance optimization of a passively designed high-rise residential building in different climatic zones of China. *Applied energy* **2018,** 215, 145-158.
[29] Chen, X.; Huang, J.; Zhang, W.; Yang, H., Exploring the optimization potential of thermal and power performance for a low-energy high-rise building. *Energy Procedia* **2019,** 158, 2469-2474.
[30] Aish, R., Bentley's Generative Components: a Design Tool for Exploratory Architecture. *Bentley Systems Inc* **2003**.
[31] Bentley, Generative Components, https://www.bentley.com/en/products/product-line/modeling-and-visualization-software/generativecomponents, Last seen: 25/2/2021. **2003**.
[32] Dynamo, Dynamo: Open source graphical programming for design, https://dynamobim.org/, Last seen: 25/2/2021. **2011**.
[33] Keough, I., Dynamo: designing a visual scripting interface for the Revit API (notes). https://github.com/ikeough/Dynamo/wiki **2011**.
[34] Rutten, D., Grasshopper3D, https://www.grasshopper3d.com/. In **2015**.
[35] Kirimtat, A.; Koyunbaba, B. K.; Chatzikonstantinou, I.; Sariyildiz, S., Review of simulation modeling for shading devices in buildings. *Renewable and Sustainable Energy Reviews* **2016,** 53, 23-49.
[36] Loh, W.-L., On Latin hypercube sampling. *The annals of statistics* **1996,** 24, (5), 2058-2080.
[37] Crombecq, K.; Dhaene, T. In *Generating sequential space-filling designs using genetic algorithms and monte carlo methods*, Asia-Pacific Conference on Simulated Evolution and Learning; Springer: **2010**; pp 80-84.
[38] Chatzikonstantinou, I.; Sariyildiz, S., Approximation of simulation-derived visual comfort indicators in office spaces: a comparative study in machine learning. *Architectural Science Review* **2016,** 59, (4), 307-322.
[39] Roman, N. D.; Bre, F.; Fachinotti, V. D.; Lamberts, R., Application and characterization of metamodels based on artificial neural networks for building performance simulation: a systematic review. *Energy and Buildings* **2020**, 109972.
[40] Grus, J., *Data science from scratch: first principles with python*. O'Reilly Media: **2019**.
[41] Hecht-Nielsen, R., Theory of the backpropagation neural network. In *Neural networks for perception*, Elsevier: **1992**; pp 65-93.
[42] Ruder, S., An overview of gradient descent optimization algorithms. *arXiv preprint arXiv:1609.04747* **2016**.
[43] Bottou, L., Large-scale machine learning with stochastic gradient descent. In *Proceedings of COMPSTAT'2010*, Springer: **2010**; pp 177-186.
[44] Kingma, D. P.; Ba, J., Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980* **2014**.
[45] Mukkamala, M. C.; Hein, M., Variants of rmsprop and adagrad with logarithmic regret bounds. *arXiv preprint arXiv:1706.05507* **2017**.
[46] De Boer, P.-T.; Kroese, D. P.; Mannor, S.; Rubinstein, R. Y., A tutorial on the cross-entropy method. *Annals of operations research* **2005,** 134, (1), 19-67.
[47] Kullback, S.; Leibler, R. A., On information and sufficiency. *The annals of mathematical statistics* **1951,** 22, (1), 79-86.
[48] Mood, A. M., Introduction to the Theory of Statistics. **1950**.
[49] Willmott, C. J.; Matsuura, K., Advantages of the mean absolute error (MAE) over the root mean square error (RMSE) in assessing average model performance. *Climate research* **2005,** 30, (1), 79-82.
[50] Draper, N. R.; Smith, H., *Applied regression analysis*. John Wiley & Sons: **1998**; Vol. 326.
[51] Srivastava, N.; Hinton, G.; Krizhevsky, A.; Sutskever, I.; Salakhutdinov, R., Dropout: a simple way to prevent neural networks from overfitting. *The journal of machine learning research* **2014,** 15, (1), 1929-1958.
[52] Stone, M., Cross-validatory choice and assessment of statistical predictions. *Journal of the Royal Statistical Society: Series B (Methodological)* **1974,** 36, (2), 111-133.
[53] Ekici, B.; Cubukcuoglu, C.; Turrin, M.; Sariyildiz, I. S., Performative computational architecture using swarm and evolutionary optimisation: A review. *Building and Environment* **2019,** 147, 356-371.
[54] Mangkuto, R. A.; Feradi, F.; Putra, R. E.; Atmodipoero, R. T.; Favero, F., Optimisation of daylight admission based on modifications of light shelf design parameters. *Journal of Building Engineering* **2018,** 18, 195-209.
[55] Vera, S.; Uribe, D.; Bustamante, W.; Molina, G., Optimization of a fixed exterior complex fenestration system considering visual comfort and energy performance criteria. *Building and Environment* **2017,** 113, 163-174.

[56] Yi, Y. K., Building facade multi-objective optimization for daylight and aesthetical perception. *Building and Environment* **2019,** 156, 178-190.
[57] Wagdy, A.; Elghazi, Y.; Abdalwahab, S.; Hassan, A., The balance between daylighting and thermal performance based on exploiting the kaleidocycle typology in hot arid climate of Aswan, Egypt. In *AEI*, **2015**; pp 300-315.
[58] Pilechiha, P.; Mahdavinejad, M.; Rahimian, F. P.; Carnemolla, P.; Seyedzadeh, S., Multi-objective optimisation framework for designing office windows: quality of view, daylight and energy efficiency. *Applied Energy* **2020,** 261, 114356.
[59] Del Ser, J.; Osaba, E.; Molina, D.; Yang, X.-S.; Salcedo-Sanz, S.; Camacho, D.; Das, S.; Suganthan, P. N.; Coello, C. A. C.; Herrera, F., Bio-inspired computation: Where we stand and what's next. *Swarm and Evolutionary Computation* **2019,** 48, 220-250.
[60] Wolpert, D. H.; Macready, W. G., No free lunch theorems for optimization. *IEEE transactions on evolutionary computation* **1997,** 1, (1), 67-82.
[61] Si, B.; Wang, J.; Yao, X.; Shi, X.; Jin, X.; Zhou, X., Multi-objective optimization design of a complex building based on an artificial neural network and performance evaluation of algorithms. *Advanced Engineering Informatics* **2019,** 40, 93-109.
[62] Cevizci, E.; Kutucu, S.; Morales-Beltran, M.; Ekici, B.; Tasgetiren, M. F., Structural Optimization for Masonry Shell Design Using Multi-objective Evolutionary Algorithms. In *Optimization in Industry*, Springer: **2019**; pp 85-119.
[63] Kim, Y.; Chung, E.-S.; Jun, S.-M.; Kim, S. U., Prioritizing the best sites for treated wastewater instream use in an urban watershed using fuzzy TOPSIS. *Resources, Conservation and Recycling* **2013,** 73, 23-32.
[64] Goussous, J.; Al-Refaie, A., Evaluation of a green building design using LCC and AHP techniques. *Life Science Journal* **2014,** 11, (8s), 29-40.
[65] Riquelme, N.; Von Lücken, C.; Baran, B. In *Performance metrics in multi-objective optimization*, Latin American Computing Conference (CLEI), 2015; IEEE: **2015**; pp 1-11.
[66] Chatzikonstantinou, I.; Sariyildiz, I. S., Addressing design preferences via auto-associative connectionist models: Application in sustainable architectural Façade design. *Automation in Construction* **2017,** 83, 108-120.
[67] USGBC, LEED v4 for building design and construction. *USGBC Inc* **2014**.
[68] Bauer, C.; Kazanasmaz, Z. T.; Wittkopf, S., Climate based daylight simulations with EvalDRC–analysis of Daylight Redirecting Components. *Journal of Facade Design and Engineering* **2017,** 5, (2), 45-57.
[69] Korsavi, S. S.; Zomorodian, Z. S.; Tahsildoost, M., Visual comfort assessment of daylit and sunlit areas: A longitudinal field survey in classrooms in Kashan, Iran. *Energy and Buildings* **2016,** 128, 305-318.
[70] Nezamdoost, A.; Van Den Wymelenberg, K.; Mahic, A., Assessing the energy and daylighting impacts of human behavior with window shades, a life-cycle comparison of manual and automated blinds. *Automation in Construction* **2018,** 92, 133-150.
[71] Nezamdoost, A.; Van Den Wymelenberg, K. G., Revisiting the daylit area: Examining daylighting performance using subjective human evaluations and simulated compliance with the LEED version 4 daylight credit. *Leukos* **2017,** 13, (2), 107-123.
[72] Sherif, A.; Sabry, H.; Wagdy, A.; Mashaly, I.; Arafa, R., Shaping the slats of hospital patient room window blinds for daylighting and external view under desert clear skies. *Solar Energy* **2016,** 133, 1-13.
[73] Wagdy, A.; Sherif, A.; Sabry, H.; Arafa, R.; Mashaly, I., Daylighting simulation for the configuration of external sun-breakers on south oriented windows of hospital patient rooms under a clear desert sky. *Solar Energy* **2017,** 149, 164-175.
[74] IES, Approved method: IES spatial Daylight autonomy (sDA) and annual sunlight exposure (ASE). **2013**.
[75] Jakubiec, J. A.; Reinhart, C. F. In *DIVA 2.0: Integrating daylight and thermal simulations using Rhinoceros 3D, Daysim and EnergyPlus*, Proceedings of building simulation, Citeseer: **2011**; pp 2202-2209.
[76] Ekici, B.; Kazanasmaz, T.; Turrin, M.; Tasgetiren, M. F.; Sariyildiz, I. S. In *A Methodology for daylight optimisation of high-rise buildings in the dense urban district using overhang length and glazing type variables with surrogate modelling*, Journal of Physics: Conference Series; IOP Publishing: **2019**; p 012133.
[77] Kirimtat, A.; Krejcar, O.; Ekici, B.; Tasgetiren, M. F., Multi-objective energy and daylight optimization of amorphous shading devices in buildings. *Solar Energy* **2019,** 185, 100-111.
[78] Lee, J.; Boubekri, M.; Liang, F., Impact of building design parameters on daylighting metrics using an analysis, prediction, and optimization approach based on statistical learning technique. *Sustainability* **2019,** 11, (5), 1474.
[79] Van Rossum, G. D., Fred L., *Python 3 Reference Manual*. CreateSpace: Scotts Valley, CA, **2009**.

[80] Pedregosa, F.; Varoquaux, G.; Gramfort, A.; Michel, V.; Thirion, B.; Grisel, O.; Blondel, M.; Prettenhofer, P.; Weiss, R.; Dubourg, V., Scikit-learn: Machine learning in Python. *the Journal of machine Learning research* **2011,** 12, 2825-2830.
[81] Chollet, F., keras, https://github.com/fchollet/keras. *GitHub repository* **2015**.
[82] Abadi, M.; Barham, P.; Chen, J.; Chen, Z.; Davis, A.; Dean, J.; Devin, M.; Ghemawat, S.; Irving, G.; Isard, M. In *Tensorflow: A system for large-scale machine learning*, 12th {USENIX} symposium on operating systems design and implementation ({OSDI} 16), **2016**; pp 265-283.
[83] McKinney, W. In *Data structures for statistical computing in python*, Proceedings of the 9th Python in Science Conference, Austin, TX: **2010**; pp 51-56.
[84] Taccari, L., Joyplots, https://github.com/sbebo/joypy/blob/master/Joyplot.ipynb. *Github repository* **2017**.
[85] Plotly, Plotly Technologies Inc., Collaborative data science, Montreal, QC, https://plotly.com/. **2015**.
[86] Hunter, J. D., Matplotlib: A 2D graphics environment. *Computing in science & engineering* **2007,** 9, (3), 90-95.
[87] Ekici, B.; Kazanasmaz, Z. T.; Turrin, M.; Tasgetiren, M. F.; Sariyildiz, I. S., Multi-zone simulation results on ASE and sDA daylight metrics for parametric high-rise model with quad grid and diagrid facade in a highly dense hypothetical urban district using dry summer climate weather data https://doi.org/10.4121/uuid:8538ac2f-3a78-4923-8fca-5beb50017299. In 4TU.ResearchData: **2020**.

	METHOD(S)	CHALLENGE(S)
	INTRODUCTION	
	LITERATURE REVIEW	
J1 **RQ1**	**Optimising form-finding parameters in performative computational architecture**	■ 17 Form-finding parameters ● 13 Performance aspects ▲ 12 Evolutionary algorithms 3 Swarm algorithms
	TOOL DEVELOPMENT AND PILOT STUDY	
J2 **RQ2**	PART A **Developing Optimus tool using self-adaptive ensemble evolutionary algorithm**	TEST 1 ■ 30 Parameters ▲ 4 Optimisation algorithms TEST 2 ■ 70 Parameters ▲ 4 Optimisation algorithms
J3 **RQ3**	PART B **Preliminary results of multi-zone approach using pilot high-rise model**	TEST 3 ■ 100 Parameters ● 2 Daylight metrics ▲ 5 ANN models 1 Optimisation algorithm
	METHODOLOGICAL FRAMEWORK	
J4 **RQ4**	PART A **Introducing multi-zone optimisation (MUZO) methodology and prediction results of quad-grid and diagrid high-rise scenarios**	TEST 4 ■ 260 Parameters ● 2 Daylight metrics ▲ 20 ANN models TEST 5 ■ 220 Parameters ● 2 Daylight metrics ▲ 20 ANN models
J5 **RQ5**	PART B **Optimising high-rise scenarios using the predictive models with Optimus and validation of the MUZO methodology**	TEST 4 ■ 260 Parameters ● 20 Predictive models ▲ 3 Optimisation algorithms TEST 5 ■ 220 Parameters ● 20 Predictive models ▲ 3 Optimisation algorithms
	CASE STUDY	
J6 **RQ6**	**Optimising Europoint complex for self-sufficiency in energy consumption and food production using MUZO and Optimus**	■ 117 Parameters ● 1 Self-sufficiency in energy 1 Self-sufficiency in food 1 Daylight metric ▲ 45 ANN models 13 Optimisation algorithms
	CONCLUSIONS	

J Journal
RQ Research question

■ Parameter ● Performance ▲ AI Method

TOC

PART B

Multi-zone optimisation of high-rise buildings using artificial intelligence for sustainable metropolises

PART 2 – Optimisation problems, algorithms, results, and method validation

ABSTRACT High-rise building optimisation is becoming increasingly relevant owing to global population growth and urbanisation trends. Previous studies have demonstrated the potential of high-rise optimisation but have been focussed on the use of the parameters of single floors for the entire design; thus, the differences related to the impact of the dense surroundings are not taken into consideration.
Part 1 of this study presents a multi-zone optimisation (MUZO) methodology and surrogate models (SMs), which provide a swift and accurate prediction for the entire building design; hence, the SMs can be used for optimisation processes. Owing to the high number of parameters involved in the design process, the optimisation task remains challenging. This paper presents how MUZO can cope with an enormous number of parameters to optimise the entire design of high-rise buildings using three algorithms with an adaptive penalty function. Two design scenarios are considered for quad-grid and diagrid shading devices, glazing type, and building-shape parameters using the setup, and the SMs developed in part 1. The optimisation part of the MUZO methodology reported satisfactory results for spatial daylight autonomy and annual sunlight exposure by meeting the Leadership in Energy and Environmental Design standards in 19 of 20 optimisation problems. To validate the impact of the methodology, optimised designs were compared with 8748 and 5832 typical quad-grid and diagrid scenarios, respectively, using the same design parameters for all floor levels. The findings indicate that the MUZO methodology provides significant improvements in the optimisation of high-rise buildings in dense urban areas.

KEYWORDS Performance-based design, building simulation, sustainability, high-rise building, machine learning, optimisation

4.7 Introduction : Part B

The demand for high-rise buildings is increasing in metropolises owing to population growth and urbanisation trends [1]. For realising sustainable urban areas, sustainable high-rise buildings should be one of the topics under investigation because they consume a significant amount of energy owing to their excessively large size [2]. Designing a sustainable high-rise building is a complex task because the process involves various types of design parameters that affect multiple performance aspects. Rafiei and Adeli [3] presented robust optimisation algorithms and neural dynamic models for investigating sustainable high-rise alternatives to cope with this complexity. The previous works mentioned in part 1 showed that optimisation algorithms and machine learning techniques have been widely used for designing sustainable high-rise buildings over the last two decades. However, in none of these studies, were the various floor levels considered as separate design problems, which is crucial for improving the overall performance of high-rise buildings [4]. Using the same design parameters for the entire high-rise design is a limited approach because the performance of the building varies between the ground and sky floor levels in dense urban areas. Optimising the design of an entire high-rise building is challenging as the simulations require expensive computational time, and the optimisation process needs to cope with an enormous number of design parameters. The use of multi-zone optimisation (MUZO) methodology is proposed to divide high-rise buildings into subdivisions (zones) to be considered as separate problems using artificial intelligence methods to address both aspects. Part 1 of the study is focussed on solving computationally expensive simulations of each zone using surrogate models (SMs). Part 2 deals with the optimisation challenge, wherein each zone is considered as a design problem using algorithms belonging to different optimisation domains. In parts 1 and 2 of the MUZO study, quad-grid and diagrid scenarios with the shading device, glazing type, and building-shape parameters were used to demonstrate the proposed methodology.

This study is focussed on optimising the entire design of high-rise buildings for quad-grid and diagrid scenarios using the 40 SMs developed in part 1. The performance aspects of the study take into consideration the two daylight metrics of Leadership in Energy and Environmental Design (LEED) v4.1., namely, the spatial daylight autonomy (sDA) and annual sunlight exposure (ASE). The optimisation process uses phase 3 of the MUZO methodology for single-objective constrained formulation with three algorithms: self-adaptive differential evolution with an ensemble of mutation strategies (jEDE) in the Optimus plug-in [5], radial basis function optimisation (RbfOpt), and covariance matrix adaptation with evolution strategy

(CMA-ES) in the Opossum plug-in [6]. In addition, an adaptive penalty function, called the near-feasibility threshold (NFT) [7,8], is used for each optimisation algorithm in the Grasshopper 3D algorithmic modelling environment (GH) [9]. The paper reports the optimisation results of 20 problems for two scenarios, which comprise 260 and 220 design parameters, respectively, with the aforementioned algorithms for five replications. Part 2 of the study also validates the significance of the proposed methodology by presenting a comparison of the performances of the optimised high-rise designs and typical high-rise scenarios generated by the same design parameters for all the floor levels. The optimisation results and validation of the method show that the MUZO methodology can play a significant role in investigating sustainable high-rise alternatives in metropolises. The rest of this paper is structured as follows: Section 4.8 presents state of the art for sDA and ASE optimisation, Section 4.9 introduces the optimisation problems and algorithms of this paper, Section 4.10 reports the optimisation results, Section 4.11 presents the validation of the MUZO methodology, Section 4.12 discusses the importance and potential of MUZO with surrogate-based design optimisation, and Section 4.13 presents the conclusions of this paper.

4.8 State of the art for sDA and ASE optimisation

This section presents the previous optimisation studies for the sDA and ASE daylight metrics of LEED within the performative computational architecture (PCA) framework in two subsections: one presenting conventional optimisation and the other computational optimisation. Conventional methods comprise an analysis of the predefined design parameters, whereas computational methods involve the use of optimisation algorithms while automating the PCA framework to investigate the best design performance. Subsequently, the novelty of this study is summarised.

4.8.1 Conventional optimisation

Over the last decade, sDA and ASE metrics have been used to investigate daylight performance and visual comfort for various building functions. An early study was focussed on a classroom case with the use of three optimisation approaches while using the optical properties and size of a south-facing window [10]. Owing to the classroom requirements, the authors maximised $sDA_{500/50\%}$ to evaluate an illuminance level of 500 lx with respect to $ASE_{1000,250h}$. In the case of a hospital-patient room, in two studies, the window blinds were optimised by shaping the slats and the configuration of external sun-breakers on south-oriented windows to maximise $sDA_{300/50\%}$ subject to $ASE_{1000,250h}$ [11,12]. In the case of office spaces, in three studies, $sDA_{300/50\%}$ was maximised subject to an $ASE_{1000,250h}$ less than 10% as a preferable result, and between 10% and 20% as an acceptable limit for various design parameters, i.e., solar screens, 3D tessellation, fixed/dynamic shading devices, and surface reflectance [13-15]. The general approach of these studies was to maximise $sDA_{300/50\%}$, with the exception of one study, owing to the educational requirements [10]. The $ASE_{1000,250h}$ was generally considered as less than 10% as a comfort limit, while two studies considered the results of less than 20% as acceptable solutions [14,15]. In addition, in the aforementioned studies, a limited number of design alternatives that might be related to conventional optimisation techniques were examined. Consequently, none of these studies were focussed on optimising the daylight performance for the design of entire buildings, such as high-rise buildings.

4.8.2 Computational optimisation

Optimisation algorithms have been widely used to cope with the complexity of the design problem while investigating desirable sDA and ASE results for various building functions. An early example was focussed on office space to maximise $sDA_{300/50\%}$ subject to an $ASE_{1000,250h}$ of less than 10% using a genetic algorithm (GA) in the Galapagos plug-in of GH while considering a single-objective formulation for kaleidocycle typology [16]. In addition to daylight, Vera, et al. [17] addressed single-objective constrained optimisation to minimise the total energy usage subject to a $sDA_{300/50\%}$ greater than 50% and $ASE_{2000,400h}$ less than 20% for exterior fenestration systems of office spaces using particle swarm optimisation with the Hooke–Jeeves algorithm in GenOpt. Another example of combining performance aspects into one fitness function was examined by Yi, et al. [18] to maximise $sDA_{300/50\%}$ and minimise $ASE_{1000,250h}$ and daylight glare probability (DGP) for auxetic structures with advanced daylight control systems in an office space using a GA in the Galapagos. As an alternative to single-objective constrained formulation, Tabadkani, et al. [19] and Mangkuto, et al. [20] maximised $|sDA - ASE|$ subject to a $sDA_{300/50\%}$ greater than 50% and 75%, and $ASE_{1000,250h}$ less than 10% for sun-responsive skin and light shelf design in office and hospital spaces using a GA in the Galapagos and Octopus plug-ins. In the case of multi-objective optimisation, Yi [21] maximised the $sDA_{300/50\%}$ and minimised $ASE_{1000,250h}$ with an aesthetic perception objective function using non-dominated sorting genetic algorithm II for a hotel building. Pilechiha, et al. [22] also considered the quality of the view from office windows in the optimisation of the $sDA_{300/50\%}$, $ASE_{1000,250h}$, and energy usage intensity while considering weighted summation and the HypE algorithm in the Octopus plug-in. As an alternative to multi-objective optimisation, Mangkuto, et al. [23] identified the simulation results for an office space with full factorial analysis of the internal shading devices to explore the non-dominated solutions while maximising $sDA_{300/50\%}$ and minimising $ASE_{1000,250h}$ and $DGP_{>0.21}$ subject to an sDA greater than 55%, ASE less than 10%, and DGP less than 50%. Five of these studies comprised the consideration of a single objective, whereas others used multi-objective and weighted summation formulations. Three studies utilised static penalty functions that might limit the search ability during the optimisation process. Finally, none of the reviewed studies consisted of a comparison of the results of different optimisation algorithms using various initial populations (replications) for the entire design of the building.

4.8.3 Novelty of this paper

This study is focussed on the optimisation of an entire high-rise building for the quad-grid and diagrid scenarios through phase 3 of the MUZO methodology, which is based on the use of multiple algorithms with replications for each optimisation task owing to the no free lunch (NFL) theorem [24]. Because of the computational burden of optimising the entire design, the high-rise building is divided into 10 subdivisions (zones), which correspond to 10 design problems starting from the first zone (Z1) at the ground level until the tenth zone (Z10) at the sky level (Fig. 4.15). Forty SMs, and the high-rise setup, which were developed in part 1 of this study, were used to optimise the sDA and ASE metrics based on the simulation results obtained for the second and fifth floors in each zone. The quad-grid scenario comprises 2.893399115e+28 design alternatives with 26 parameters, whereas this number is 3.054543465e+23 for the diagrid scenario with 22 parameters in one zone. For each optimisation task, phase 3 of the MUZO methodology is considered by employing the jEDE, RbfOpt, CMA-ES algorithms, and NFT adaptive penalty function with five replications, which suggests a decision-making process using 15 optimisation results. Consequently, this paper reports on the optimised high-rise buildings after a total of 300 optimisation runs is complete, using 260 parameters for the quad-grid, and 220 parameters for diagrid, and it validates the impact of the proposed methodology by comparing the optimised scenarios with the typical high-rise scenarios. Thus, part 2 of the study not only deals with the optimisation of the entire design of high-rise buildings for the performance metrics under study, but also addresses 20 complex design problems, each having an enormous number of design alternatives in the optimisation search space, owing to the involvement of multiple design parameters.

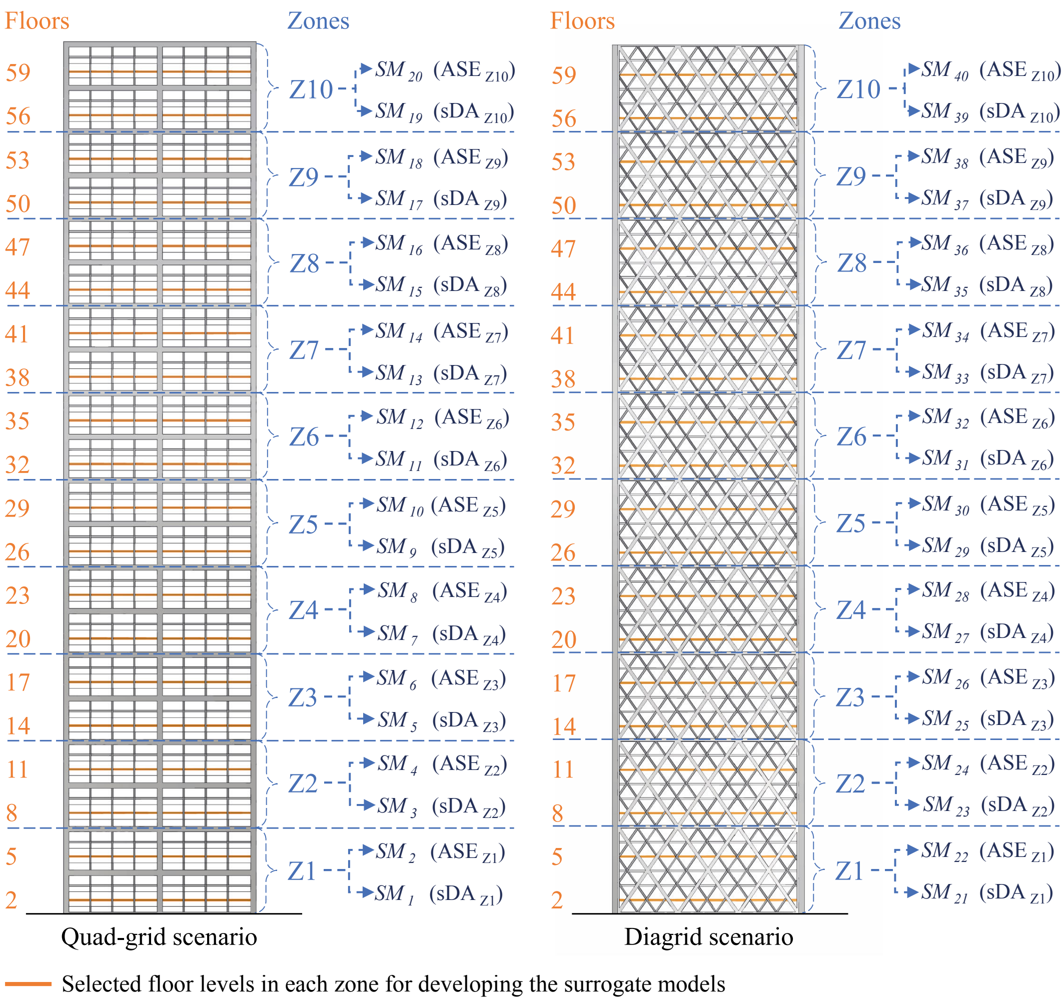


FIG. 4.15 Subdivisions (zones) of high-rise scenarios and their surrogate models

TOC

4.9 Optimisation problems and algorithms

This section explains the problem formulation and algorithms used in each optimisation process. The first subsection explains the single-objective constrained formulation, whereas the subsequent subsections present the RbfOpt, CMA-ES, and jEDE algorithms with applications in the architecture domain. Finally, the NFT describes the adaptive penalty function for constraint handling.

4.9.1 Problem formulation

The Illuminating Engineering Society (IES) recommends a minimum $sDA_{300/50\%}$ of 55% with a maximum $ASE_{1000,250h}$ of 10% as desirable daylight with acceptable comfort [25]. However, the LEED standards acknowledge design proposals with two points, i.e., while the $sDA_{300/50\%}$ is greater than 55% and $ASE_{1000,250h}$ is less than 10% for regularly occupied floor areas. When reaching a minimum of 75% of $sDA_{300/50\%}$ with 10% of $ASE_{1000,250h}$, the design is acknowledged with three points. Considering the formulations of previous studies and the recommendation of the IES and the LEED standards, in this study, a single-objective constrained optimisation is considered for each design problem as

$$\begin{array}{lll} \max: & sDA_{300/50\%} & X=(x_1,x_2,\ldots,x_n) \;\; and \;\; X \in S \\ subject\ to: & ASE_{1000/250h} \leq ASE_{bound} & \end{array} , \qquad (4.12)$$

where n is the number of design parameters in each zone for both quad-grid and diagrid scenarios, S is the entire search space of one zone, and ASE_{bound} is the maximum limit for direct sunlight. State of the art shows that the ASE results can be related by more than 10% to the design of the shading devices. Because the sufficiency of shading devices is unexplored at the beginning of the optimisation processes, an adaptive ASE boundary is considered in each zone as

$$ASE_{bound} = \begin{cases} 10\% \;\; if \;\; sDA_{300/50\%} \geq 55\% \\ 20\% \;\; if \;\; sDA_{300/50\%} \leq 55\% \;\; and \;\; ASE_{1000/250h} > 10\% \\ 30\% \;\; if \;\; sDA_{300/50\%} \leq 55\% \;\; and \;\; ASE_{1000/250h} > 20\% \end{cases} , \qquad (4.13)$$

where ASE_{bound} increases by 10% when the sDA result is less than 55%. This approach is considered in both quad-grid and diagrid scenarios to optimise the sDA and ASE metrics using the SMs. The optimisation task starts from Z1 and ends at Z10. After the best parameter set is determined in one zone for each algorithm, the optimisation process of the next zone is started. The parameters presented in part 1 of the MUZO study are also used herein (Table 4.1). The supplementary material presents the predictive models with learning scores of 40 SMs that were used during the optimisation process.

4.9.2 Radial basis function optimisation

RbfOpt is a model-based algorithm used for solving computationally expensive problems and was recently presented by Costa and Nannicini [26]. For the unknown cost function, the algorithm constructs and iteratively refines an approximation model with sampled points. Compared to the existing open-source model-based algorithms available, RbfOpt provides two main contributions: an efficient method for automatic model selection using a cross-validation scheme, and an approach to exploit noisy but faster function evaluations. Opossum provides the RbfOpt algorithm to be used in architectural design optimisation as an open-source plug-in developed for GH [6]. RbfOpt in Opossum has been widely used for various design problems, i.e., daylight and glare problems [27], optimal viewing angle in stadium design [28], structural optimisation [29], urban design [30], and optimisation problems focussed on building energy [31]. In this study, the optimisation process uses the default RbfOpt parameters while running the algorithm through Opossum v2.0.0.

4.9.3 Covariance matrix adaptation with evolution strategy

CMA-ES is a well-known optimisation algorithm in the evolutionary computation (EC) domain proposed by [32-34]. One of its most powerful features is that the search space can be increased or decreased in the next iteration based on the results of every solution. The algorithm uses this procedure for the multivariate normal distribution parameters (mean and sigma) and for the entire covariance matrix that belongs to the decision variable space. Opossum v1.7.0 provides a CMA-ES algorithm for design optimisation in the architecture domain as an open-source plug-in for GH. Recently, this algorithm has been used for various design problems, e.g., Waibel, Wortmann, Evins and Carmeliet [31] optimised building energy problems while reporting promising results with a large evaluation budget, Zhang, et al. [35]

focussed on aerodynamic shape optimisation problems, and Fortich Mora [36] used CMA-ES for the design problem of sustainable high-rise buildings. The optimisation process in this study comprises the use of Opossum v2.0.0, while considering the default features of the CMA-ES algorithm.

4.9.4 Self-adaptive differential evolution with ensemble of mutation strategies

jEDE is a hybrid algorithm that belongs to the EC domain using differential evolution [37], self-adaptive strategy [38], and an ensemble of mutation strategies [39]. The purpose of the algorithm is to cope with high-dimensional problems in the domain of architectural design optimisation. The algorithm comprises a self-adaptive approach that converges to different directions with various rates of mutation and crossover operators. Moreover, with the ensemble idea, jEDE also selects the best mutation strategy for every dimension among predefined operators during the optimisation process. Therefore, the algorithm can adapt its search behaviour to different problems. The first application of jEDE, which is provided by Optimus v1.0.0 as an open-source plug-in for GH, was used for 30D CEC 2005 benchmark problems and a 70D structural design problem [5]. The algorithm presented promising results as compared with particle swarm optimisation, genetic algorithm, and RbfOpt. In addition, recent publications have demonstrated the potential of jEDE in solving a 20D problem of daylight [40] and the optimisation of sustainable high-rise building design focussed on daylight, comfort, and energy use intensity aspects with SMs [36]. The optimisation process in this study comprises the use of the default parameters of Optimus v1.0.2 for the jEDE algorithm.

4.9.5 Near feasibility threshold constraint handling

In previous studies mentioned in section 4.8, single-objective constrained optimisation is considered as a problem formulation for the ASE and sDA metrics according to the LEED and IES standards. The general approach of these studies was to consider the ASE as a constant penalty function to be embedded in the sDA fitness function. In this method, the result of the fitness function (sDA) is multiplied with a constant value if the solution of the constraint function (ASE) is in the infeasible region. Previous studies have also discussed that the ASE results could be related to the sufficiency of the shading devices by more than 10%. Another reason for this outcome may be related to the limited searchability of the constant penalty functions. In the case of challenging constraint problems, Mallipeddi and Suganthan [41] emphasised the importance of using advanced constraint-handling approaches. Therefore, in this study, the NFT adaptive penalty function is taken into consideration [7], which is an advanced version of the constant penalty function. The approach of the NFT is to define a threshold distance from a feasible region and to encourage the search within this region and the NFT neighbourhood while discouraging the search beyond that threshold. Equation (4.14) and Equation (4.15) explain the penalised fitness function $f_p(x)$ using the NFT as

$$f_p(x) = f(x) + \left(\frac{v(x)}{NFT}\right)^{\alpha}, \quad (4.14)$$

$$NFT = \frac{NFT_0}{1+\lambda \cdot g}, \quad (4.15)$$

where $f(x)$ is the fitness function; $v(x)$ is the violation; α and λ are user-defined positive parameters taken as 2 and 0.04, respectively, NFT_0 is the upper bound of the NFT taken as 0.1; and g is the generation or iteration number. The optimisation process of RbfOpt, CMA-ES, and jEDE takes into consideration the NFT approach to obtain a reasonable comparison between algorithms for each problem. The Optimus plug-in v1.0.2 provides an open-source NFT module that can work with other optimisation plug-ins in GH.

4.10 Results

The optimisation results were obtained using a computer with an Intel Xeon E5-1620 v3 core processor at 3.50 GHz, 16-GB DDR3 memory, and a 512-GB solid-state drive (Fig. 4.16). As the termination criterion, 10,000 was considered as the maximum number of function evaluations (*FES*). In the implementation of CMA-ES and RbfOpt, non-populated approaches were considered in the Opossum plug-in. Therefore, 10,000 was set as the maximum *FES* for CMA-ES and RbfOpt, while 40 population sizes and 250 generations were considered for the population-based jEDE algorithm. During the optimisation process, Opossum automatically stopped the iteration if there was no alteration in the fitness function. Therefore, the computation times of all the algorithms were also recorded. To evaluate the

TOC

optimisation performance of RbfOpt, CMA-ES, and jEDE, the following five criteria were considered: *max f(x)*, and *std f(x)*, respectively, are the maximum, and standard deviation of the function x for five replications; *CPU* is the average time in seconds to complete one replication; *FES* is the total number of function evaluations, and *FES/CPU* is the number of completed function evaluations in 1 s (Fig. 4.17). The convergence graphs for the best results among the five replications of each algorithm are presented in Figs. 4.18 and 4.19. In addition, appendices 4C, 4D, 4E and 4F present the convergence graphs of all the replications.

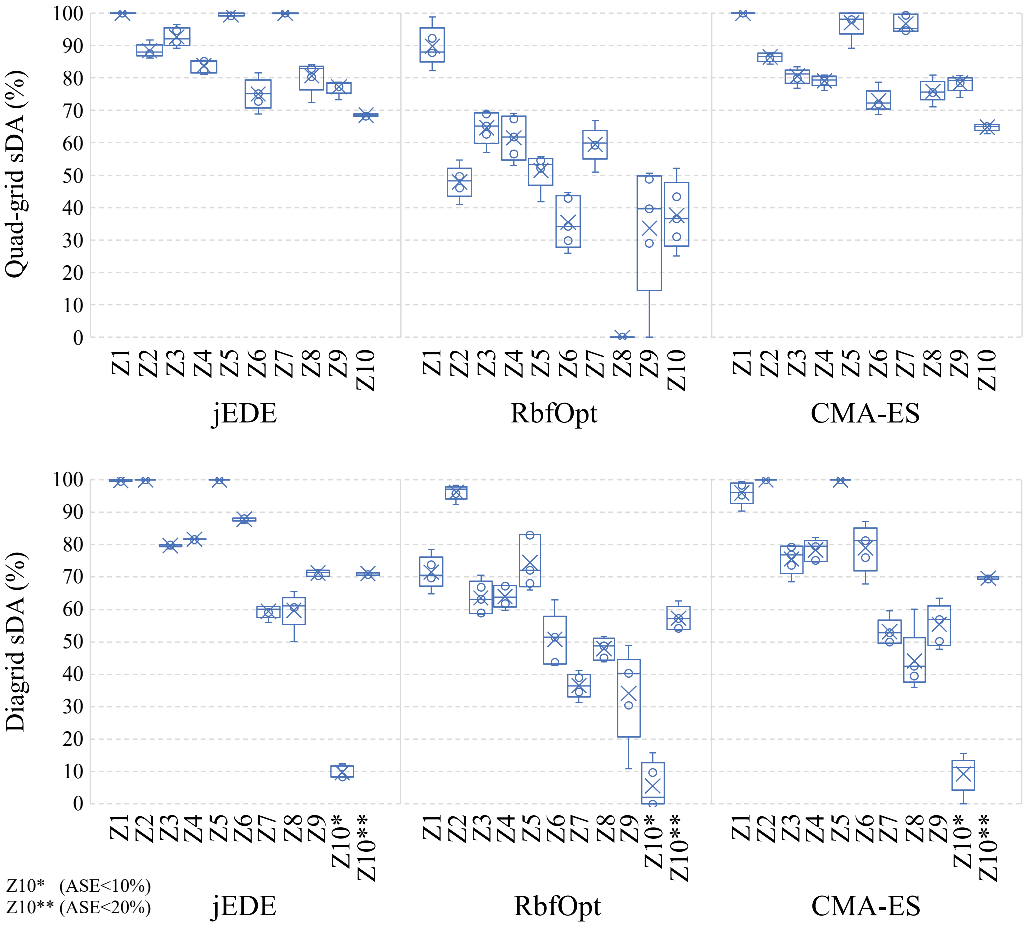


FIG. 4.16 Boxplots of the optimisation results

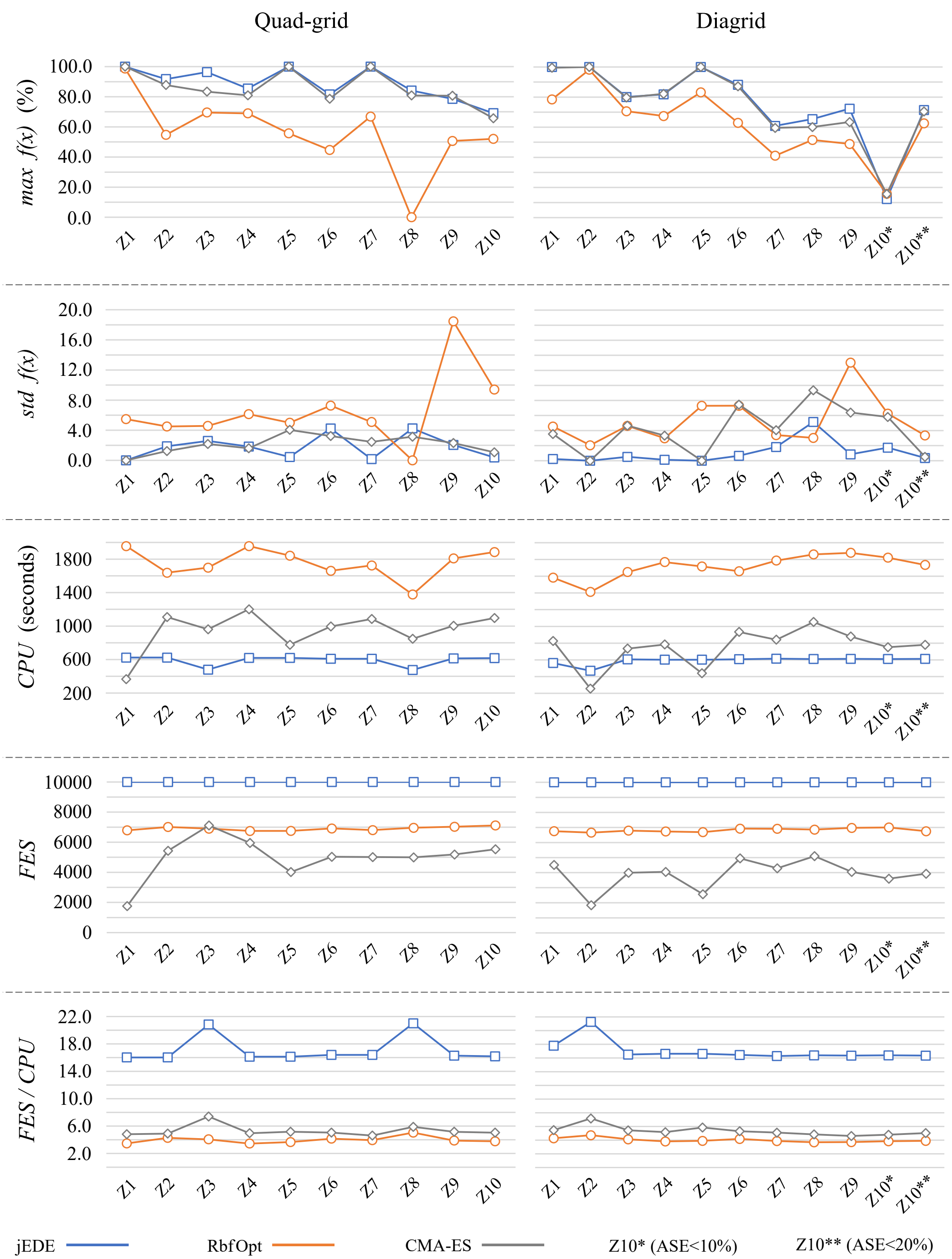


FIG. 4.17 *max f(x)*, *std f(x)*, *CPU*, *FES*, and *FES/CPU* for five replications

In the quad-grid results, jEDE outperformed the other algorithms in six zones, whereas jEDE and CMA-ES yielded the same results in three zones, and CMA-ES outperformed jEDE in one zone. In the case of the LEED scores, jEDE and CMA-ES reached three points in nine zones, while both algorithms reached two points only in Z10.

In contrast, RbfOpt reported three points for Z1, two points for Z3, Z4, Z5, and Z7, and sDA results less than 55% in other zones. Hence, the jEDE and CMA-ES could cope with the quad-grid scenario and provided satisfactory results for LEED standards, while the RbfOpt could not achieve the same result owing to the insufficient sDA levels reported for Z2, Z6, Z8, Z9, and Z10. In the diagrid results, the constraint of $ASE_{bound} \leq 10$ resulted in undesirable sDA solutions in Z10 for all the algorithms. Thus, the boundary was increased by 10% to consider the new constraint function as $ASE_{bound} \leq 20$. As a result, the jEDE outperformed the other algorithms in seven zones. In two zones, jEDE and CMA-ES yielded the same results, whereas only in one zone, the CMA-ES outperformed the jEDE.

In the case of the LEED scores, the jEDE and CMA-ES presented three points in six zones and two points in three zones, whereas the RbfOpt found three points in three zones, two points in three zones, and insufficient results in four zones. Therefore, the jEDE and CMA-ES could cope with the diagrid scenario, while providing satisfactory results for the LEED standards in nine zones and acceptable results ($ASE_{bound} \leq 20$) in Z10, while the RbfOpt could not present a desirable performance for the entire building owing to the insufficient sDA results reported for Z7, Z8, Z9, and Z10. With respect to the computation time, the RbfOpt and CMA-ES were automatically terminated at a smaller *FES* than the jEDE. Based on the *CPU* results, the CMA-ES converged faster than the other algorithms in Z1 of the quad-grid, and Z2 and Z5 of the diagrid scenarios. In all the other problems, the jEDE converged faster than the CMA-ES and RbfOpt with less deviation in computation time despite the higher *FES*. In contrast, the *FES/CPU* results suggested that the jEDE could evaluate a single function much faster than the other algorithms.

In the optimised solutions, the results showed that the sDA values diversified in all zones for both scenarios. For instance, optimised solutions of the lower zones presented a high percentage of sDA because the dense areas in the built environment significantly blocked direct sunlight. Thus, the daylight was controlled using shading devices and considering high-transmittance glazing materials between Z1 and Z3. In the middle zones, it was observed that the sDA values started to vary between Z4 and Z7 owing to the different shading densities and glazing types used.

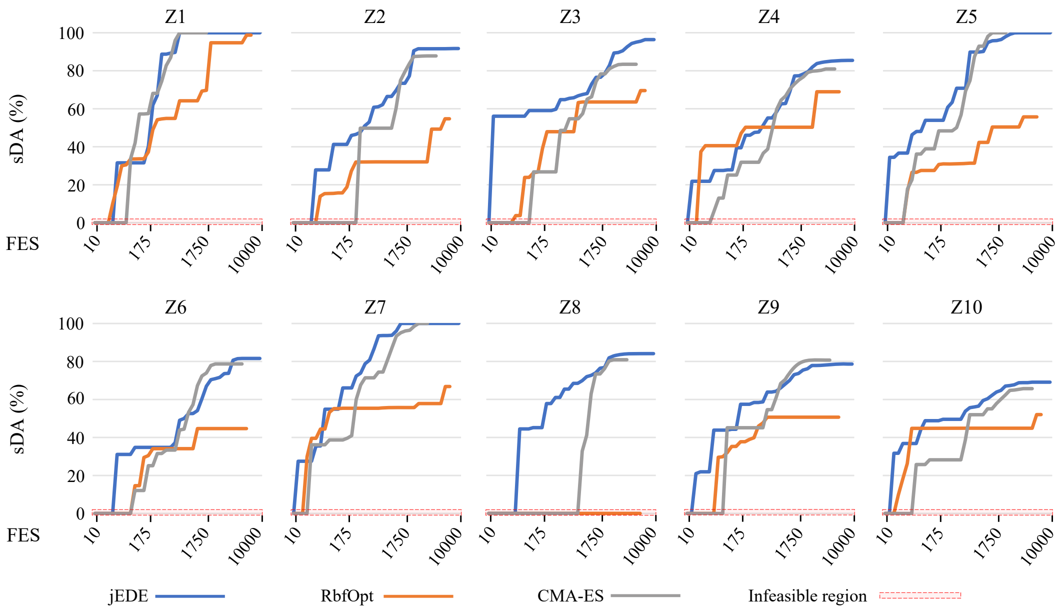


FIG. 4.18 Convergence graphs of the best optimisation results for the quad-grid scenario

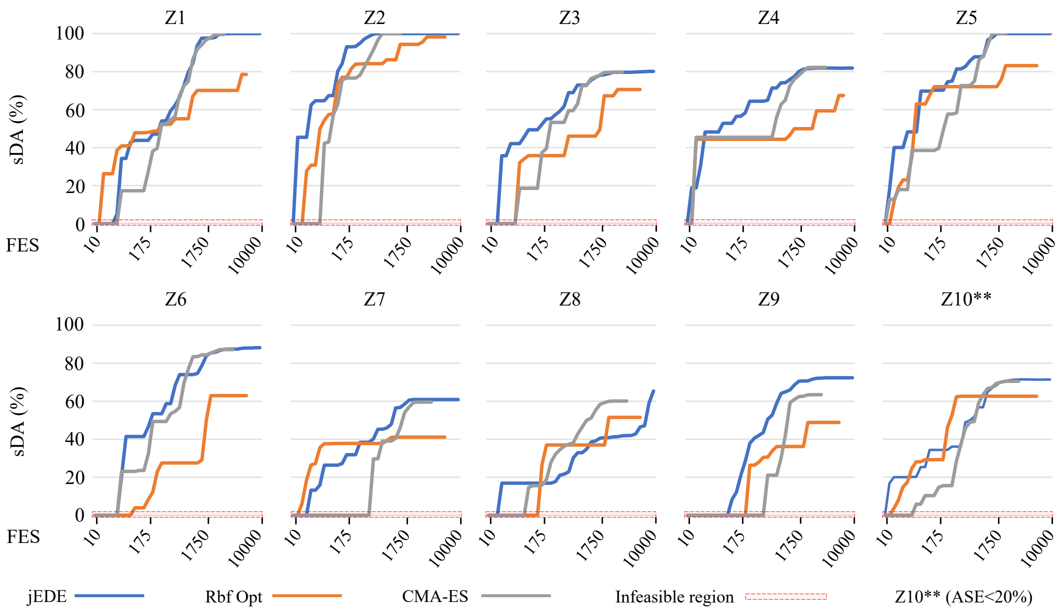


FIG. 4.19 Convergence graphs of the best optimisation results for the diagrid scenario

In the higher zones (Z8–Z10), the sDA results were lower than those in the other zones because direct sunlight met with the corresponding floors from all directions (north, south, east, and west). Therefore, either dense use of shading devices or low-transmittance glazing materials were selected, especially in the south and east orientations, to cope with this challenge. In addition, it was observed that a significant building twist would be desirable in the zones between Z8 to Z10 to decrease the impact of direct sunlight as compared with the other zones. The described design differences in the various zones were based on several reasons. Firstly, the density of the surroundings caused various design challenges, i.e., high building density at the ground levels and low density at the sky levels. Therefore, the optimisation algorithms found different design parameters owing to the different surrounding conditions. Secondly, higher zones were dependent on the lower zones because of the rotation and floor-to-floor height parameters. The optimised parameters in the lower zones could negatively affect the higher zones. Nevertheless, desirable solutions were obtained from the results reported after the MUZO optimisation process because the independent rotation and floor-to-floor height parameters could control the performance of each zone.

With a focus on the overall building performance based on the average results of all the zones, Table 4.4 presents the sDA results for the entire high-rise building. The overall results of the algorithms demonstrated that the jEDE and CMA-ES found a higher sDA in the quad-grid than in the diagrid. However, the RbfOpt presented a superior sDA performance in the diagrid scenario. Consequently, the jEDE presented the best sDA performance, while the CMA-ES presented the second-best performance, and the RbfOpt presented the third best design options. Moreover, based on the results in Figs. 4.16 to 4.19, we can also conclude that the quad-grid shading devices provided better daylight performance within acceptable comfort conditions as compared with the diagrid devices. Fig. 4.20 presents the best parameters reported after the optimisation process for both scenarios, whereas Figs. 4.21 and 4.22 illustrate these parameters in the form of high-rise buildings. The supplementary material presents the results of the optimised building designs.

TABLE 4.4 sDA performance of the entire high-rise building design for quad-grid and diagrid scenarios

Algorithm	Quad-grid	Diagrid
jEDE	**88.7**	**82.0**
RbfOpt	56.2	66.5
CMA-ES	85.8	80.2

	jEDE										RbfOpt										CMA-ES									
Quad-grid	Z1	Z2	Z3	Z4	Z5	Z6	Z7	Z8	Z9	Z10	Z1	Z2	Z3	Z4	Z5	Z6	Z7	Z8	Z9	Z10	Z1	Z2	Z3	Z4	Z5	Z6	Z7	Z8	Z9	Z10
x1,…,x10	4.9	4.7	4.9	5.0	4.8	4.8	4.8	4.3	4.0	4.4	4.9	4.2	4.4	4.0	4.1	4.2	4.2	4.0	4.1	4.0	4.9	4.1	5.0	4.5	4.8	4.6	4.7	4.0	4.0	4.3
x11,…,x20	-10	-10	-3	2	10	-10	-10	-10	-10	-10	-10	-10	-5	0	-10	-10	-10	-5	-9	-10	-8	-10	-6	3	10	-10	-10	-10	-10	-10
Q1	4	7	7	0	0	0	0	8	0	8	3	4	3	0	8	0	1	8	0	1	8	7	6	0	1	0	0	5	0	4
Q2	1.0	1.5	0.0	0.0	0.0	0.0	0.3	1.5	0.6	0.0	0.4	1.5	0.7	1.0	0.1	0.0	0.5	1.5	0.0	0.2	0.1	1.5	0.0	0.1	0.2	0.0	0.0	1.3	0.5	0.0
Q3	-60	-60	-59	-60	57	60	60	60	-53	-60	-57	23	-40	-32	19	56	52	-6	-57	-25	-60	-60	-60	-60	60	16	59	59	-25	-60
Q4	0	0	0	0	0	2	0	0	0	0	1	0	0	0	0	0	0	0	2	0	0	0	0	0	0	2	0	0	0	0
Q5	0.0	0.1	1.5	1.5	1.5	0.0	1.4	1.5	0.0	1.5	0.1	0.0	1.0	0.8	1.5	0.9	0.9	0.0	0.6	1.1	0.9	0.0	1.5	1.5	1.5	0.0	1.5	1.5	0.2	1.5
Q6	0	7	0	6	0	2	7	3	0	0	0	0	0	5	3	0	4	8	4	0	0	8	0	0	0	3	8	0	0	0
Q7	1.5	0.2	0.0	1.5	0.0	0.0	1.5	0.0	0.0	0.0	1.5	0.0	0.0	0.3	0.3	0.1	0.4	1.1	1.2	0.0	1.4	0.0	0.0	1.5	0.1	0.0	1.5	0.0	0.0	0.0
Q8	60	-57	60	-60	60	60	-60	60	-60	-58	59	-56	54	19	56	59	-21	-59	58	15	48	-49	50	-60	58	60	-48	60	-60	-58
Q9	2	0	2	0	2	2	2	0	2	2	2	0	2	0	1	2	2	2	2	2	2	0	2	0	2	2	2	0	2	2
Q10	1.1	1.5	1.5	0.0	1.5	1.5	0.0	0.0	0.0	0.0	1.5	0.7	1.4	0.0	0.0	1.3	1.2	1.1	0.4	1.4	0.9	1.5	1.5	0.0	1.5	1.5	0.1	0.0	0.0	0.0
Q11	0	8	0	8	0	8	1	8	2	8	0	8	0	8	4	8	3	7	5	8	0	8	0	7	6	8	0	8	0	8
Q12	0.0	0.0	0.3	1.0	0.0	0.0	0.0	0.0	0.0	0.0	1.2	0.2	0.8	0.0	0.0	0.4	0.2	1.0	0.1	0.9	0.9	0.0	0.0	0.5	0.0	0.0	0.0	0.0	0.0	0.0
Q13	60	58	-57	-58	53	60	59	-55	-60	54	-5	-29	-47	-7	43	-41	57	-36	-20	59	40	-60	-60	-5	58	-22	51	-57	-60	12
Q14	2	2	0	2	2	2	2	2	2	1	2	2	0	2	2	2	2	2	0	2	2	2	0	2	2	2	2	2	2	2
Q15	1.3	1.9	0.3	1.5	1.5	1.5	1.5	1.5	1.5	0.0	1.5	1.5	0.1	1.5	1.5	1.5	1.5	1.5	0.0	0.7	1.5	1.5	0.0	1.5	1.5	1.5	1.5	1.5	1.5	0.0
Q16	8	0	8	0	8	8	3	8	8	8	8	6	4	0	8	8	3	8	6	8	0	0	8	0	8	8	5	8	8	8
Q17	0.0	1.4	1.5	0.0	0.0	1.5	1.5	0.0	1.1	0.0	0.6	1.4	1.2	0.0	0.1	1.2	0.3	0.5	1.5	1.1	0.9	1.5	1.0	0.0	0.0	1.5	1.3	0.0	1.2	0.0
Q18	37	-60	58	-60	-51	-60	60	-59	-59	-55	-20	-56	46	-59	48	-59	40	40	-8	-60	41	-60	58	-60	-14	-60	60	-31	-12	-30
Q19	2	2	2	0	2	2	2	0	0	2	2	2	2	0	2	2	2	2	2	2	2	2	2	0	2	2	2	0	0	2
Q20	1.4	1.5	1.5	0.0	1.5	1.5	1.5	0.0	1.5	1.5	0.2	1.5	1.5	0.1	1.5	1.5	1.5	0.1	1.3	0.0	1.0	1.5	1.5	0.0	1.5	1.5	1.5	0.4	1.5	1.5
Q21	4	4	4	4	4	4	4	4	4	4	4	2	4	4	4	4	4	3	4	4	4	4	4	4	4	4	4	4	4	4
Q22	4	4	4	1	2	4	4	4	4	4	4	4	1	3	1	2	3	2	1	4	3	4	1	2	3	4	4	4	4	4
Q23	3	2	4	1	4	4	4	4	4	1	2	2	4	2	4	2	4	1	1	1	4	4	4	1	4	4	4	4	4	1
Q24	4	4	4	4	4	4	3	2	3	4	4	3	4	4	4	3	2	1	4	4	4	4	4	4	4	4	3	2	3	4
Diagrid	Z1	Z2	Z3	Z4	Z5	Z6	Z7	Z8	Z9	Z10	Z1	Z2	Z3	Z4	Z5	Z6	Z7	Z8	Z9	Z10	Z1	Z2	Z3	Z4	Z5	Z6	Z7	Z8	Z9	Z10
x1,…,x10	4.4	4.0	4.0	4.0	4.0	4.0	4.0	4.0	4.1	4.0	4.0	4.2	4.0	4.0	4.0	4.0	4.0	4.0	4.0	4.0	4.4	4.0	4.0	4.0	4.0	4.0	4.0	4.0	4.0	4.0
x11,…,x20	-10	9	-10	-10	7	10	-10	-9	-8	-9	-6	5	-10	-9	6	0	-5	7	-10	-7	-10	8	-7	-10	6	10	-10	10	-9	-8
D1	0.0	0.2	0.5	0.0	1.5	1.5	0.0	0.3	1.5	1.5	0.0	0.6	1.5	0.9	1.4	1.0	0.0	1.3	1.3	1.4	0.0	0.1	0.3	0.0	1.4	1.5	0.0	0.5	1.4	0.4
D2	0.0	0.1	0.0	1.5	0.0	0.0	1.5	0.0	0.0	0.0	0.1	0.0	0.0	1.4	0.0	1.0	0.9	0.0	0.2	0.0	0.0	0.4	0.0	1.5	0.0	0.4	0.0	0.3	0.0	0.0
D3	-60	-60	57	-60	-59	-60	-46	-60	-60	-60	-9	16	40	59	-53	-58	-26	-7	59	18	-60	0	-8	-60	-41	-60	-60	-60	-60	55
D4	0	0	0	0	0	0	0	0	0	0	0	1	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
D5	0.0	0.0	0.0	1.5	1.5	0.0	1.3	0.0	0.0	0.0	0.4	1.3	0.2	1.3	0.8	0.5	1.1	0.2	0.1	1.0	0.0	0.4	0.0	1.5	1.5	0.0	0.5	0.0	0.0	0.1
D6	1.5	0.1	1.5	0.0	0.4	1.5	0.8	0.0	0.0	0.0	1.4	0.0	1.1	0.0	0.5	1.5	1.4	0.0	1.4	0.2	1.2	1.4	1.3	0.0	0.0	1.3	0.7	0.0	0.0	0.0
D7	-60	-60	-42	-60	-60	-60	54	-58	-60	-60	-33	-50	-59	-42	45	-41	36	54	48	-59	-60	-56	-59	-60	-51	-60	20	-47	-60	-59
D8	5	2	2	2	5	5	5	5	5	0	5	5	5	0	5	4	5	5	4	3	5	2	4	2	4	5	5	3	5	0
D9	0.0	1.4	0.0	1.5	1.5	1.5	1.5	1.5	0.2	1.5	0.0	0.8	0.0	1.0	1.4	0.0	1.4	0.1	1.5	1.3	0.0	1.5	0.0	1.5	1.5	1.5	1.5	1.1	1.5	0.6
D10	0.0	1.5	0.0	0.7	0.0	1.5	1.5	0.7	0.0	0.3	0.0	1.5	1.2	1.2	0.5	0.3	1.5	0.1	0.0	0.1	0.0	1.0	0.0	0.7	0.0	1.5	1.5	0.0	1.3	0.3
D11	-60	-57	-60	-13	-60	57	-60	-34	-60	-59	-6	-38	-60	21	-60	9	32	-12	-59	-49	57	-20	-60	-13	9	60	-60	-37	-60	-60
D12	3	0	5	0	1	2	5	2	5	5	0	0	3	4	3	5	5	4	3	5	5	0	5	0	2	3	5	3	5	5
D13	1.5	1.5	1.5	0.0	1.5	1.5	1.5	0.0	1.5	0.0	1.5	1.5	1.3	0.4	1.5	1.5	1.4	0.4	1.4	0.2	1.5	1.5	1.5	0.0	1.5	1.5	1.5	0.0	1.5	0.0
D14	0.3	0.0	1.4	0.0	0.8	0.2	0.0	0.0	0.0	0.7	1.3	0.0	1.3	0.1	1.0	1.2	0.5	0.1	0.7	0.0	0.0	0.0	1.1	0.0	1.0	0.1	0.0	0.9	0.0	0.8
D15	2	16	-60	-60	-59	-44	-60	-60	59	-60	-39	23	-60	-41	17	15	-40	-59	59	-40	-57	40	-60	-60	-51	-5	-58	-60	-50	-60
D16	5	5	5	0	5	5	0	0	5	0	5	4	5	0	5	5	0	0	5	0	5	4	5	0	5	5	0	0	5	0
D17	4	4	4	4	4	4	4	4	4	4	4	4	4	3	4	4	4	4	4	3	4	4	4	4	4	4	4	4	4	4
D18	4	4	4	4	4	4	1	2	4	4	4	4	4	4	4	4	2	1	2	4	4	4	4	4	4	4	3	1	4	4
D19	4	4	4	1	3	4	4	1	1	1	4	4	4	1	4	4	4	1	1	1	4	4	4	1	3	4	4	1	1	1
D20	4	4	4	3	4	4	4	4	4	4	4	4	4	3	4	4	1	4	4	4	4	4	4	3	4	4	1	4	4	4

Legend: Minimum value | Maximum value | Minimum value | Maximum value | Minimum value | Maximum value

FIG. 4.20 Parameter maps of the optimised building designs

TOC

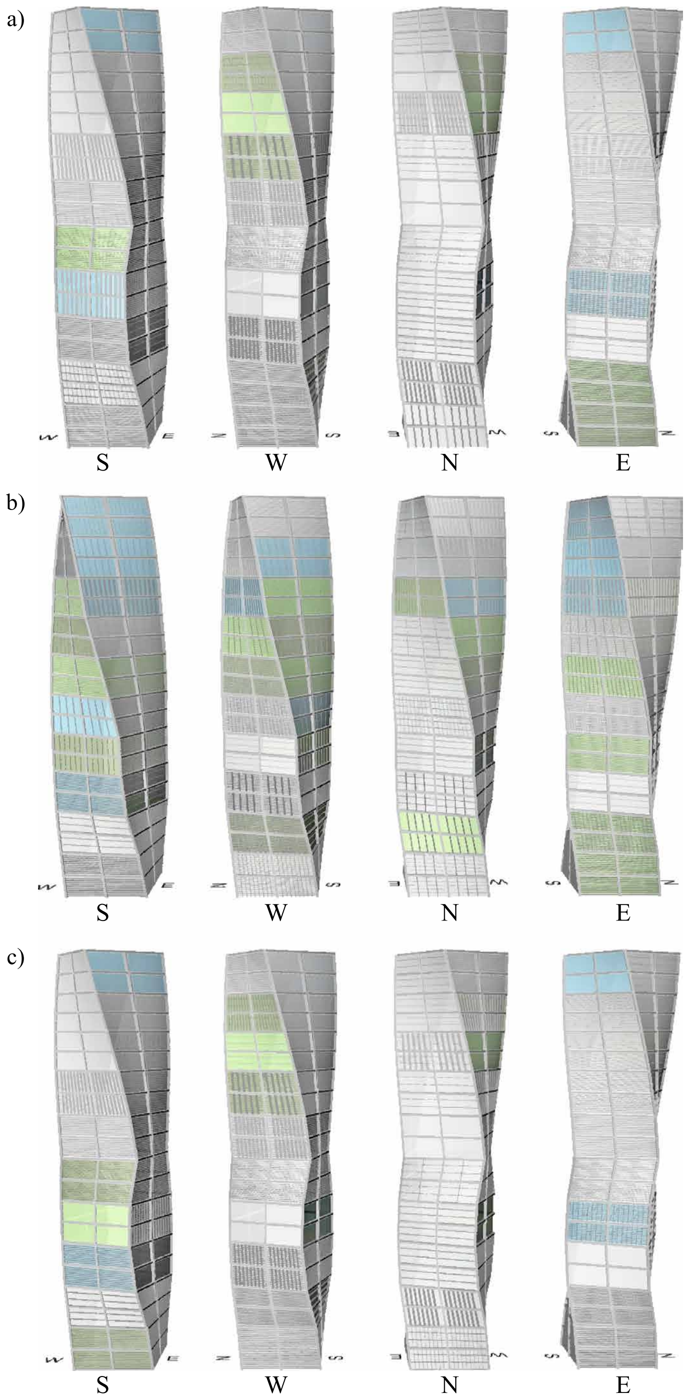


FIG. 4.21 jEDE (a), RbfOpt (b), and CMA-ES (c) optimised designs for the quad-grid scenario

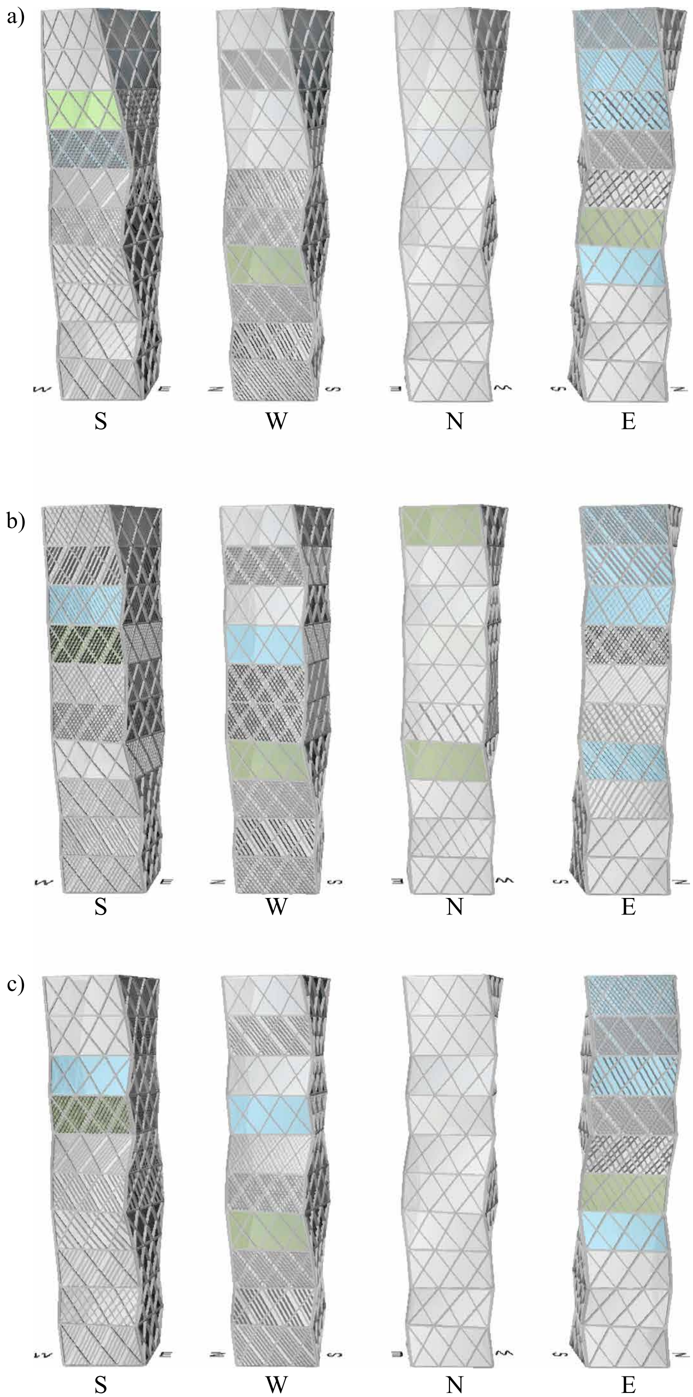


FIG. 4.22 jEDE (a), RbfOpt (b), and CMA-ES (c) optimised designs for the diagrid scenario

4.11 Validation of the method

The design of high-rise buildings has changed owing to technological improvements, design concerns with environmental impacts, and regulation changes over time [42]. Few buildings appear to be examples of such design concerns in the 21st century, as they comprise various building shapes and façade configurations and a combination of transparent and opaque surfaces. However, the design of high-rises using various design parameters could provide solutions for realising better building performance in dense urban districts, as discussed in this paper. This section presents the potential performance improvement that can be realised in sustainable high-rise buildings in metropolises by comparing the optimised scenarios obtained using the MUZO methodology with typical high-rise scenarios. In the majority of the existing high-rise buildings, the same parameter values are applied to the entire high-rise design (e.g., singular floor-to-floor height, same façade configuration, and a single glazing type). For profound comparisons, various combinations of parameters are defined to develop typical scenarios using the same parameters in the optimisation process. In total, 8748 typical quad-grid and 5832 typical diagrid scenarios were generated using the values listed in Table 4.5, and Figs. 4.23 and 4.24 illustrate several examples of these scenarios.

The performance of each typical scenario was calculated for every zone using the same SMs in a short time. The average performance results obtained for all the zones

TABLE 4.5 Parameter values used for generating typical scenarios

Scenario	Floor-to-floor height of zones	Rotation of zones	Number of the shading devices			Length of the shading devices				Rotation of shading devices	Glazing type	Generated scenarios
			Horizontal	Vertical	Diagonal	Horizontal	Vertical	1st order diagonal	2nd order diagonal			
Quad-grid	[4, 4.5, 5]	[0, 4, 8]	[0, 1, 2]	[0, 4, 8]	-	[0.0, 0.8, 1.5]	[0.0, 0.8, 1.5]	-	-	[-30, 0, 30]	[1, 2, 3, 4]	8748
Diagrid	[4, 4.5, 5]	[0, 4, 8]	-	-	[0, 1, 2, 3, 4, 5]	-	-	[0.0, 0.8, 1.5]	[0.0, 0.8, 1.5]	[-30, 0, 30]	[1, 2, 3, 4]	5832

were considered to evaluate the overall building performance for the typical scenarios. In the case of the optimised scenarios, the jEDE results were used for comparison, as they were the best-proposed design solutions. Figs. 4.25 and 4.26 present comparisons of the quad-grid and diagrid scenarios, respectively. As a result, the MUZO designs exhibited the best performances with an ASE of 9.8% and sDA of 88.7% in the quad-grid scenario and an ASE of 10.5% and sDA of 82.0% in the diagrid scenario. As mentioned in the results section, owing to the insufficient shading performance of diagrid Z10, $ASE_{bound} \leq 20$ was considered, which resulted in a slightly higher ASE performance than 10%. Ultimately, the overall performances of the typical high-rise scenarios could not provide satisfactory LEED scores, which demonstrates the importance of using the MUZO methodology in dense urban districts.

4.12 Discussion

This section presents the discussion based on the optimisation results, and the validation of the method explained in the previous sections. Firstly, two discussion topics are addressed: the importance of the MUZO methodology for metropolises and its potential. Secondly, the ongoing discussion in architectural design optimisation based on surrogate-based algorithms versus optimisation with SMs is focussed upon.

1 The importance of the MUZO methodology for future metropolises

The results obtained in this study indicated that the MUZO methodology could present desirable performance outcomes for 20 complex design problems while considering multiple parameters related to the architecture of high-rise buildings. Recent reviews have shown that not only performance aspects related to sustainable buildings but also parameters related to architectural design may present additional complexity during the optimisation process [43-46]. Therefore, the use of the MUZO methodology may support architects and engineers as they investigate sustainable high-rise scenarios by taking into consideration parameters related to design concerns in the conceptual phase. The results also proved that the performance outcomes on different floor levels of high-rise buildings may be affected in dense urban areas.

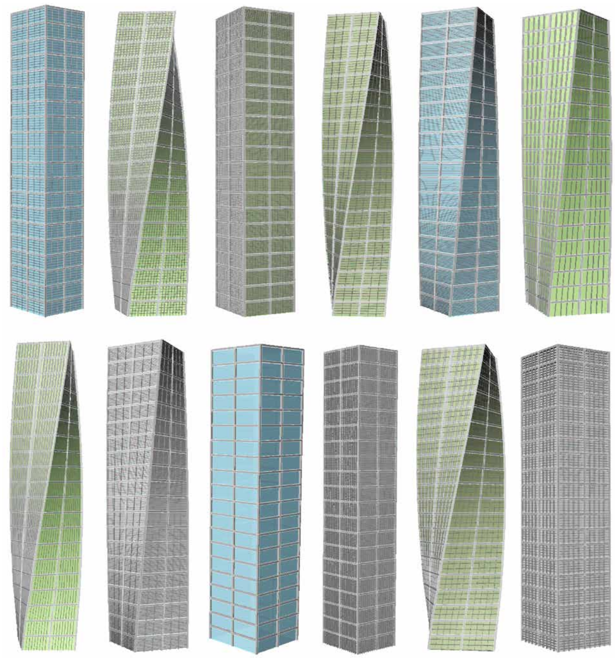


FIG. 4.23 Typical quad-grid high-rise examples

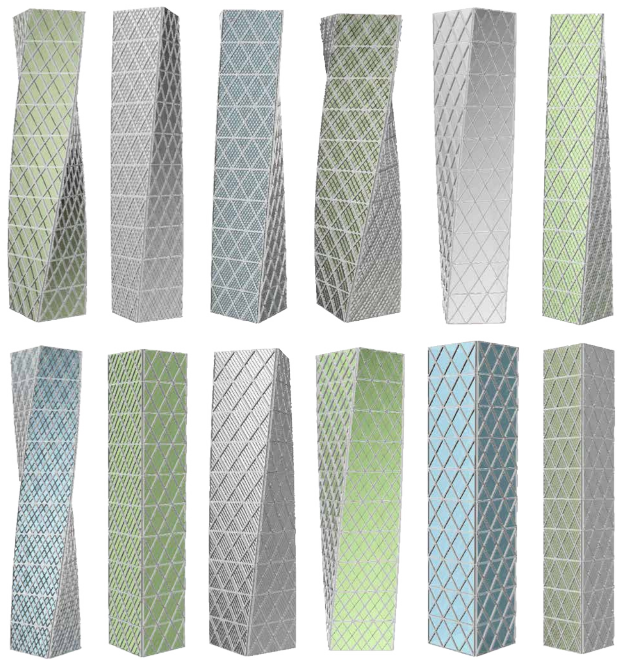


FIG. 4.24 Typical diagrid high-rise examples

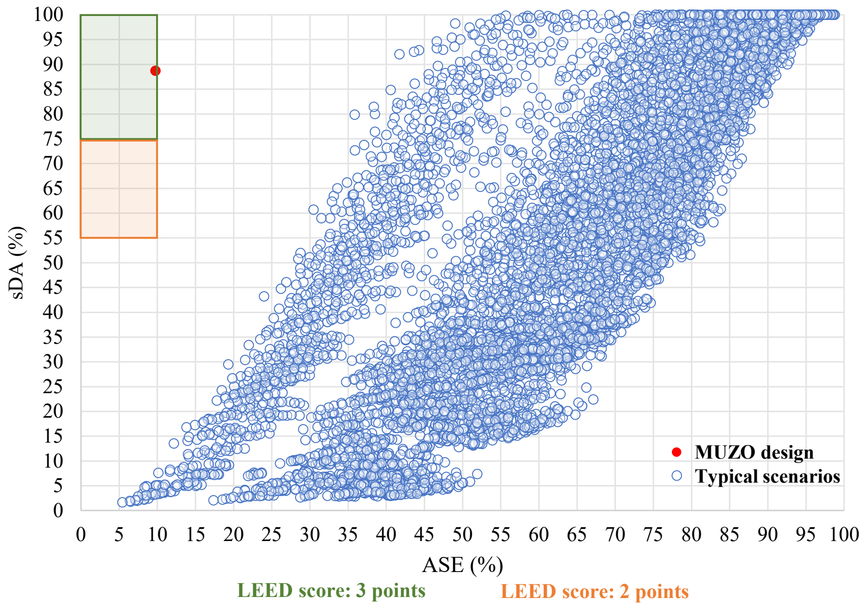


FIG. 4.25 Validation for quad-grid scenario (MUZO design versus 8748 scenarios)

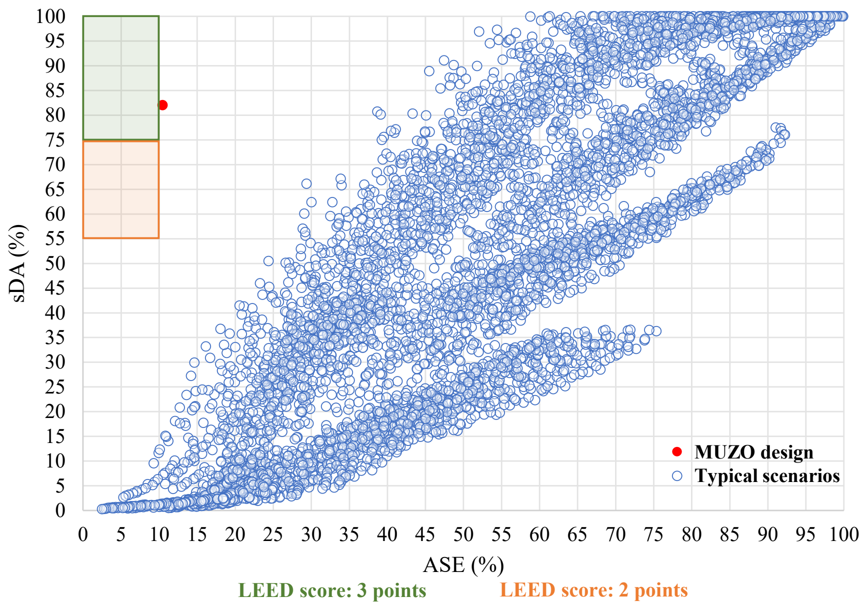


FIG. 4.26 Validation for diagrid scenario (MUZO design versus 5832 scenarios)

The main reason for the superior results obtained in the optimised designs proposed by the MUZO methodology was the division of one large design problem into sub-problems (zones). Hence, the optimisation algorithms could determine the best design alternatives for each zone while considering the performances of the various floor levels.

2 Potential of the MUZO methodology

This study focussed on optimising the sDA and ASE daylight metrics of LEED standards to evaluate the sustainability score of high-rise buildings. The MUZO methodology may integrate more performance aspects related to sustainable buildings (e.g., energy consumption, building-integrated photovoltaics, and adaptive comfort). In such a case, the formulation of the problem could comprise multi-objective or many-objective optimisation to handle more than two conflicting performance aspects. In addition, the complexity of the problem can be controlled by varying the number of zones. In this study, ten zones were considered, which is a predefined parameter that can be changed by the decision-maker based on the density of the surroundings. The consideration of fewer zones would limit the number of design decisions for the entire high-rise design, while the use of a larger number of zones may increase the complexity and computational burden exponentially. During the optimisation process, 1,095,395 and 1,139,785 *FES* were considered for the quad-grid and diagrid scenarios, respectively, in order to determine which presents the best performance, and 14,580 *FES* were considered to evaluate the performance generated in typical high-rise scenarios. If these tasks were based on simulations, which required 4 min to calculate the performance of one design, 17.12 years would be required to complete all these computations. The MUZO methodology provided near-optimal alternatives for 4 days using SMs. Moreover, the aforementioned optimisation tasks were completed in GH using the Optimus and Opossum plug-ins. The flexibility of the proposed methodology allows the use of other digital platforms for optimisation, e.g., Python, C++, and C#, because the predictive models can be defined in another software.

3 Surrogate-based optimisation algorithms versus optimisation with SMs

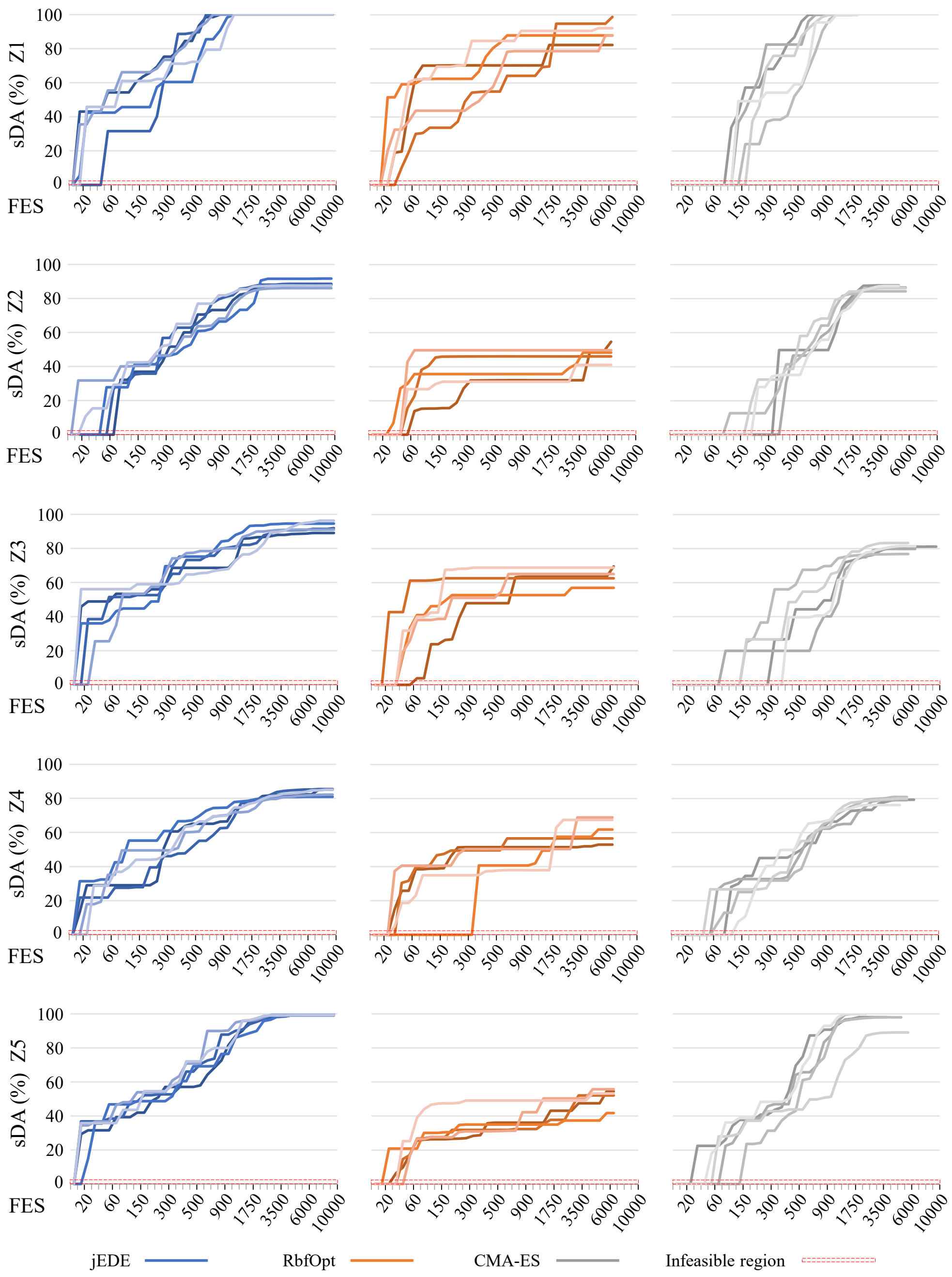
An ongoing discussion in the literature is focussed on the use of either surrogate-based optimisation (e.g., RbfOpt) or SMs with heuristic optimisation algorithms (e.g., this study). While the user can optimise the design task using surrogate-based algorithms when considering a small amount of *FES*, the overall process still requires a significant amount of time owing to the replication of the optimisation process using simulations. However, decision-makers can investigate the design problem extensively in a reasonable amount of time using SMs, various algorithms, and replications, but with a prediction error. The accuracy of the SMs can be improved for each design problem, as explained in part 1 of this study; however, achieving zero error is almost impossible. Therefore, we can conclude that surrogate-based optimisation is convenient for small-scale design problems (e.g., office spaces), whereas optimisation with SMs is useful for large-scale design problems (e.g., high-rise buildings).

4.13 Conclusion

This paper presents the second part of the MUZO study and is focussed on the optimisation problems and algorithms, results, and validation of the method. The results of this study showed that the performance of the entire high-rise building in dense urban districts can be improved by focusing on each zone as a separate design problem, and the optimisation process is explained in this paper. The combination of these approaches with the SMs presented in part 1 allowed us to complete the optimisations of entire high-rise buildings in a short time. The obtained results indicated satisfactory sDA and ASE performances that met the LEED criteria in 19 out of 20 design problems comprising various complexities. Although the jEDE slightly outperformed the CMA-ES algorithm, the RbfOpt presented a lower sDA performance as compared to the other algorithms. This underscores the importance of employing various optimisation algorithms with replications in architectural design optimisation because *the global optimal of each design problem is unexplored*. In addition, the validation of the method also demonstrated that the building performance achieved using the MUZO methodology exhibited a remarkable improvement as compared to that of typical high-rise scenarios in dense urban districts. Therefore, the consideration of different parameters for various floor levels may provide significant performance improvements in the design of sustainable high-rise buildings in metropolises.

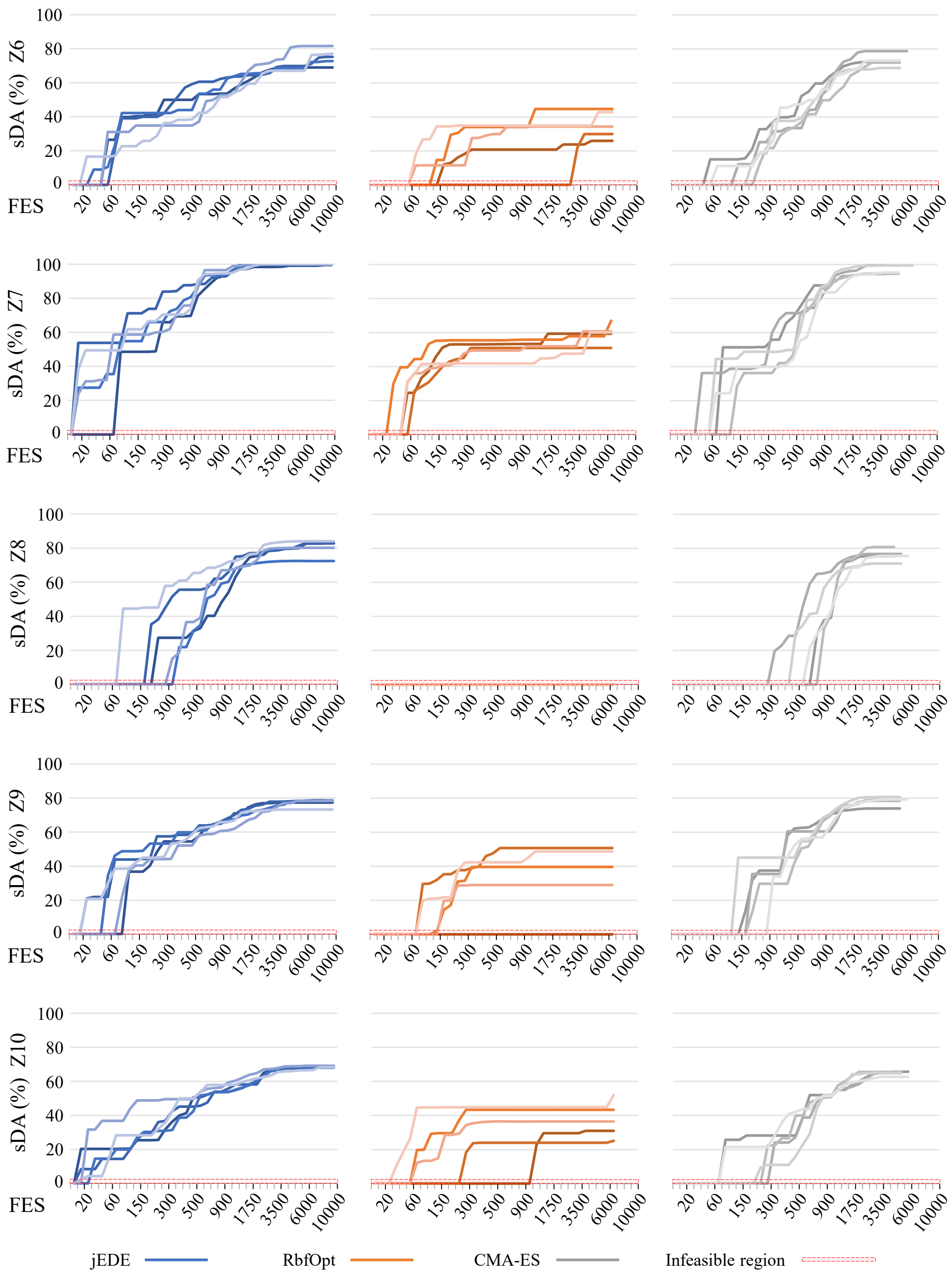
In conclusion, the relevance of this study is confirmed by the obtained optimisation results and the validation of the presented method. Thus, this study underscores the effect of the use of the MUZO approach for metropolises while dividing high-rise buildings into zones to be considered as separate design problems. The importance of artificial intelligence methods for swift optimisation for determining sustainable high-rise alternatives with the use of a large number of parameters was also demonstrated. In real-world applications, there is a possibility of combining 10 zones into one objective function instead of dealing with 10 separate problems. However, the design process may involve a large number of parameters, such as 260 parameters in the quad-grid and 220 parameters in the diagrid scenarios of this study. Therefore, the domain of architectural design optimisation requires tools and algorithms that can simultaneously cope with more than 200 parameters for high-dimensional constrained problems [47,48]. A sensitivity analysis could decrease the total number of design parameters; however, the final design may not reflect all the architectural concerns owing to some variables having been discarded. Another alternative to decrease the overall complexity of the design process could be the consideration of two algorithms that belong to different optimisation domains (e.g., surrogate-based and EC). Finally, in real-world applications, fewer zones may be considered, which would also decrease the computational complexity, based on the density of the urban plot under study.

Appendix 4C. Quad-grid convergence graphs for all replications from Z1 to Z5



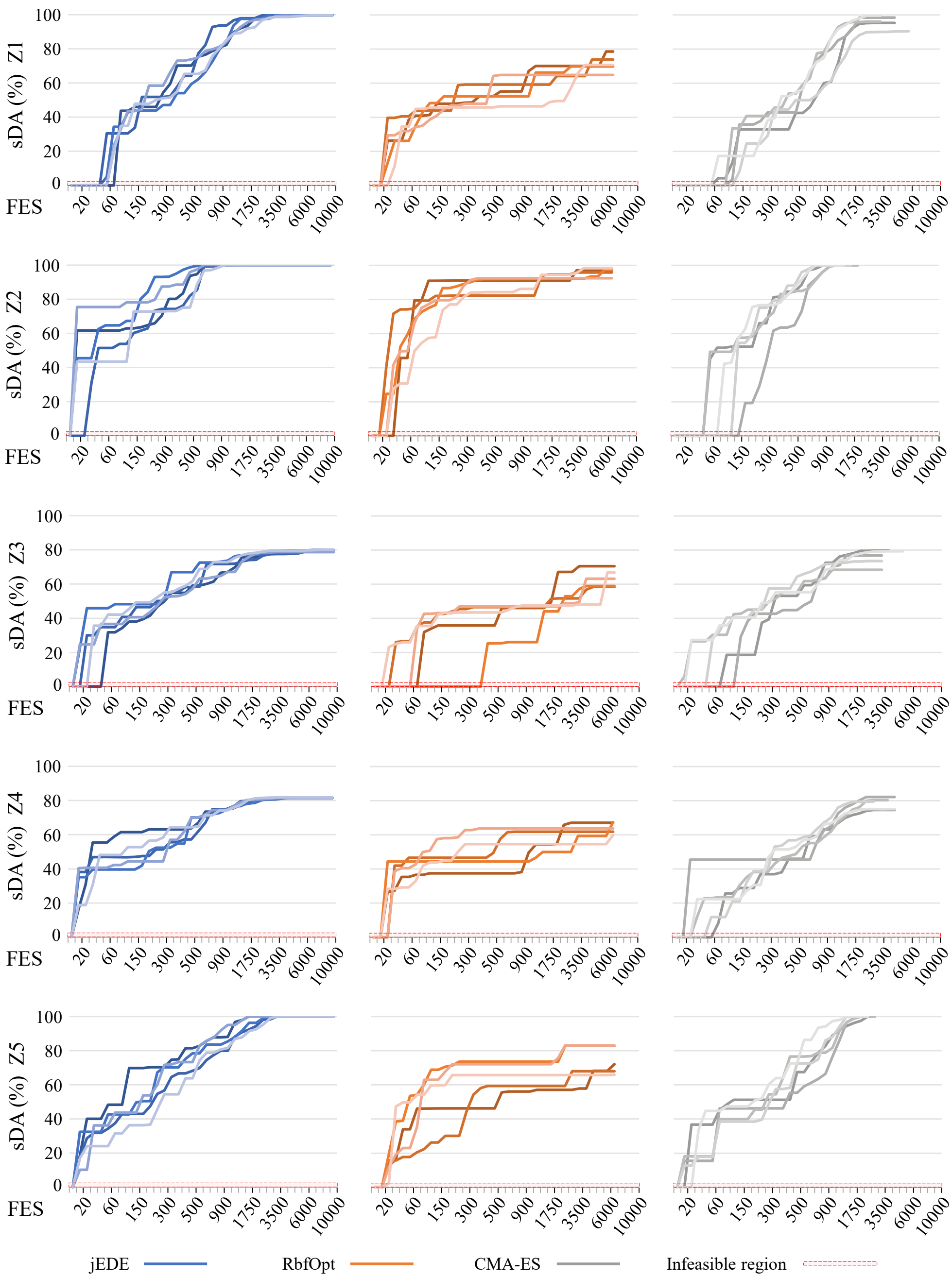
TOC

Appendix 4D. Quad-grid convergence graphs for all replications from Z6 to Z10



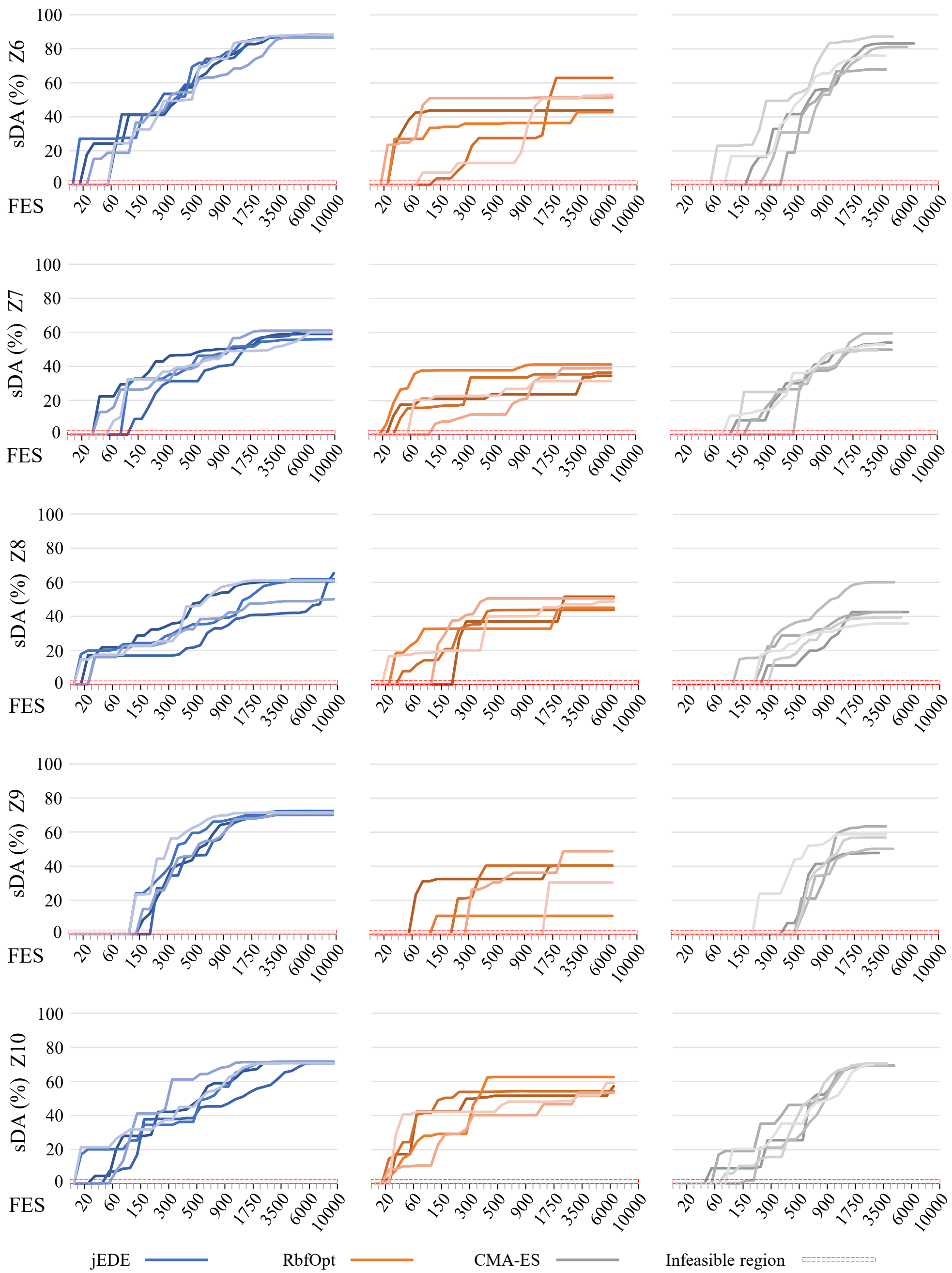
TOC

Appendix 4E. Diagrid convergence graphs for all replications from Z1 to Z5



TOC

Appendix 4F. Diagrid convergence graphs for all replications from Z6 to Z10



References

[1] Ali, M. M.; Al-Kodmany, K., Tall buildings and urban habitat of the 21st century: a global perspective. *Buildings* **2012,** 2, (4), 384-423.

[2] Ali, M. M.; Armstrong, P. J. In *Overview of sustainable design factors in high-rise buildings*, Proc. of the CTBUH 8th World Congress, **2008**; pp 3-5.

[3] Rafiei, M. H.; Adeli, H., Sustainability in highrise building design and construction. *The Structural Design of Tall and Special Buildings* **2016,** 25, (13), 643-658.

[4] Wood, A., Sustainability: a new high-rise vernacular? *The structural design of tall and special buildings* **2007,** 16, (4), 401-410.

[5] Cubukcuoglu, C.; Ekici, B.; Tasgetiren, M. F.; Sariyildiz, S., OPTIMUS: Self-Adaptive Differential Evolution with Ensemble of Mutation Strategies for Grasshopper Algorithmic Modeling. *Algorithms* **2019,** 12, (7), 141.

[6] Wortmann, T., Opossum-introducing and evaluating a model-based optimization tool for grasshopper. **2017**.

[7] Coit, D. W.; Smith, A. E., Penalty guided genetic search for reliability design optimization. *Computers & industrial engineering* **1996,** 30, (4), 895-904.

[8] Smith, A. E.; Coit, D. W., Penalty functions. *Handbook of evolutionary computation* **1997,** 97, (1), C5.

[9] Rutten, D., Grasshopper3D, https://www.grasshopper3d.com/. In **2015**.

[10] Kazanasmaz, T.; Grobe, L. O.; Bauer, C.; Krehel, M.; Wittkopf, S., Three approaches to optimize optical properties and size of a South-facing window for spatial Daylight Autonomy. *Building and Environment* **2016,** 102, 243-256.

[11] Sherif, A.; Sabry, H.; Wagdy, A.; Mashaly, I.; Arafa, R., Shaping the slats of hospital patient room window blinds for daylighting and external view under desert clear skies. *Solar Energy* **2016,** 133, 1-13.

[12] Wagdy, A.; Sherif, A.; Sabry, H.; Arafa, R.; Mashaly, I., Daylighting simulation for the configuration of external sun-breakers on south oriented windows of hospital patient rooms under a clear desert sky. *Solar Energy* **2017,** 149, 164-175.

[13] Fathy, F.; Sabry, H.; Faggal, A. A., External Versus Internal Solar Screen: Simulation Analysis for Optimal Daylighting and Energy Savings in an Office Space. *Proceedings of the PLEA, Edinburgh, UK* **2017,** 16.

[14] Giostra, S.; Masera, G.; Pesenti, M.; Pavesi, P. In *Use of 3D tessellation in curtain wall facades to improve visual comfort and energy production in buildings*, IOP Conference Series: Earth and Environmental Science, **2019**; IOP Publishing: p 012044.

[15] Palarino, C.; Piderit, M. B., Optimisation of Passive Solar Design Strategies in Side-lit Offices: Maximising Daylight Penetration While Reducing the Risk of Glare in Different Chilean Climate Contexts. *Journal of Daylighting* **2020,** 7, (1), 107-121.

[16] Wagdy, A.; Elghazi, Y.; Abdalwahab, S.; Hassan, A., The balance between daylighting and thermal performance based on exploiting the kaleidocycle typology in hot arid climate of Aswan, Egypt. In *AEI* **2015**; pp 300-315.

[17] Vera, S.; Uribe, D.; Bustamante, W.; Molina, G., Optimization of a fixed exterior complex fenestration system considering visual comfort and energy performance criteria. *Building and Environment* **2017,** 113, 163-174.

[18] Yi, Y. K.; Sharston, R.; Barakat, D., Auxetic structures and advanced daylight control systems. *Journal of Facade Design and Engineering* **2018,** 7, (1), 63-74.

[19] Tabadkani, A.; Banihashemi, S.; Hosseini, M. R. In *Daylighting and visual comfort of oriental sun responsive skins: A parametric analysis*, Building simulation; Springer: **2018**; pp 663-676.

[20] Mangkuto, R. A.; Feradi, F.; Putra, R. E.; Atmodipoero, R. T.; Favero, F., Optimisation of daylight admission based on modifications of light shelf design parameters. *Journal of Building Engineering* **2018,** 18, 195-209.

[21] Yi, Y. K., Building facade multi-objective optimization for daylight and aesthetical perception. *Building and Environment* **2019,** 156, 178-190.

[22] Pilechiha, P.; Mahdavinejad, M.; Rahimian, F. P.; Carnemolla, P.; Seyedzadeh, S., Multi-objective optimisation framework for designing office windows: quality of view, daylight and energy efficiency. *Applied Energy* **2020,** 261, 114356.

[23] Mangkuto, R. A.; Dewi, D. K.; Herwandani, A. A.; Koerniawan, M. D., Design optimisation of internal shading device in multiple scenarios: Case study in Bandung, Indonesia. *Journal of Building Engineering* **2019,** 24, 100745.

[24] Wolpert, D. H.; Macready, W. G., No free lunch theorems for optimization. *IEEE transactions on evolutionary computation* **1997,** 1, (1), 67-82.

[25] IES, Approved method: IES spatial Daylight autonomy (sDA) and annual sunlight exposure (ASE). **2013**.

[26] Costa, A.; Nannicini, G., RBFOpt: an open-source library for black-box optimization with costly function evaluations. *Mathematical Programming Computation* **2018,** 10, (4), 597-629.
[27] Wortmann, T., Model-based optimization for architectural design: Optimizing daylight and glare in grasshopper. *Technology| Architecture+ Design* **2017,** 1, (2), 176-185.
[28] Zargar, S. H.; Alaghmandan, M., CORAL: introducing a fully computational plug-in for stadium design and optimization; a case study of finding optimal spectators' viewing angle. *Architectural Science Review* **2019,** 62, (2), 160-170.
[29] Ilunga, G.; Leitão, A. In *Derivative-free Methods for Structural Optimization*, Proceedings of the 36th eCAADe Conference, Lodz, Poland, **2018**; pp 19-21.
[30] Wortmann, T.; Natanian, J., Multi-Objective Optimization for Zero-Energy Urban Design in China: A Benchmark. *Proc. SimAUD2020* **2020**, 203-210.
[31] Waibel, C.; Wortmann, T.; Evins, R.; Carmeliet, J., Building energy optimization: An extensive benchmark of global search algorithms. *Energy and Buildings* **2019,** 187, 218-240.
[32] Hansen, N., The CMA evolution strategy: a comparing review. In *Towards a new evolutionary computation*, Springer: **2006**; pp 75-102.
[33] Hansen, N.; Müller, S. D.; Koumoutsakos, P., Reducing the time complexity of the derandomized evolution strategy with covariance matrix adaptation (CMA-ES). *Evolutionary computation* **2003,** 11, (1), 1-18.
[34] Hansen, N.; Ostermeier, A., Completely derandomized self-adaptation in evolution strategies. *Evolutionary computation* **2001,** 9, (2), 159-195.
[35] Zhang, R.; Waibel, C.; Wortmann, T. In *Aerodynamic Shape Optimization for High-Rise Conceptual Design: Integrating and validating parametric design,(fast) fluid dynamics, structural analysis and optimization*, Anthropologic–Architecture and Fabrication in the cognitive age. Proceedings of the 38th International Online Conference on Education and Research in Computer Aided Architectural Design in Europe, Berlin, Germany, 16th-17th September **2020**, eCAADe (Education and Research in Computer Aided Architectural Design in Europe): 2020; pp 37-45.
[36] Fortich Mora, F. E. Humble Giants: Computational Intelligence for Designing More Sustainable High-rise Buildings using Surrogate Models, http://resolver.tudelft.nl/uuid:d6d3d3e6-6424-4650-88e2-8d741112257f. 2020.
[37] Storn, R.; Price, K., Differential evolution–a simple and efficient heuristic for global optimization over continuous spaces. *Journal of global optimization* **1997,** 11, (4), 341-359.
[38] Brest, J.; Greiner, S.; Boskovic, B.; Mernik, M.; Zumer, V., Self-adapting control parameters in differential evolution: A comparative study on numerical benchmark problems. *IEEE transactions on evolutionary computation* **2006,** 10, (6), 646-657.
[39] Mallipeddi, R.; Suganthan, P. N.; Pan, Q.-K.; Tasgetiren, M. F., Differential evolution algorithm with ensemble of parameters and mutation strategies. *Applied soft computing* **2011,** 11, (2), 1679-1696.
[40] Ekici, B.; Kazanasmaz, T.; Turrin, M.; Tasgetiren, M. F.; Sariyildiz, I. S. In *A Methodology for daylight optimisation of high-rise buildings in the dense urban district using overhang length and glazing type variables with surrogate modelling*, Journal of Physics: Conference Series, **2019**; IOP Publishing: 2019; p 012133.
[41] Mallipeddi, R.; Suganthan, P. N., Ensemble of constraint handling techniques. *IEEE Transactions on Evolutionary Computation* **2010,** 14, (4), 561-579.
[42] Oldfield, P.; Trabucco, D.; Wood, A., Five energy generations of tall buildings: an historical analysis of energy consumption in high-rise buildings. *The Journal of Architecture* **2009,** 14, (5), 591-613.
[43] Attia, S.; Hamdy, M.; O'Brien, W.; Carlucci, S., Assessing gaps and needs for integrating building performance optimization tools in net zero energy buildings design. *Energy and Buildings* **2013,** 60, 110-124.
[44] Ekici, B.; Cubukcuoglu, C.; Turrin, M.; Sariyildiz, I. S., Performative computational architecture using swarm and evolutionary optimisation: A review. *Building and Environment* **2019,** 147, 356-371.
[45] Evins, R., A review of computational optimisation methods applied to sustainable building design. *Renewable and sustainable energy reviews* **2013,** 22, 230-245.
[46] Touloupaki, E.; Theodosiou, T., Performance simulation integrated in parametric 3D modeling as a method for early stage design optimization—A review. *Energies* **2017,** 10, (5), 637.
[47] Chu, W.; Gao, X.; Sorooshian, S., A new evolutionary search strategy for global optimization of high-dimensional problems. *Information Sciences* **2011,** 181, (22), 4909-4927.
[48] Jia, D.; Zheng, G.; Qu, B.; Khan, M. K., A hybrid particle swarm optimization algorithm for high-dimensional problems. *Computers & Industrial Engineering* **2011,** 61, (4), 1117-1122.

	METHOD(S)	CHALLENGE(S)	
	INTRODUCTION		
	LITERATURE REVIEW		
J1 **RQ1**	**Optimising form-finding parameters in performative computational architecture**	■ 17 Form-finding parameters ● 13 Performance aspects ▲ 12 Evolutionary algorithms 3 Swarm algorithms	
	TOOL DEVELOPMENT AND PILOT STUDY		
J2 **RQ2**	PART A **Developing Optimus tool using self-adaptive ensemble evolutionary algorithm**	■ 30 Parameters ▲ 4 Optimisation algorithms	TEST 1
		■ 70 Parameters ▲ 4 Optimisation algorithms	TEST 2
J3 **RQ3**	PART B **Preliminary results of multi-zone approach using pilot high-rise model**	■ 100 Parameters ● 2 Daylight metrics ▲ 5 ANN models 1 Optimisation algorithm	TEST 3
	METHODOLOGICAL FRAMEWORK		
J4 **RQ4**	PART A **Introducing multi-zone optimisation (MUZO) methodology and prediction results of quad-grid and diagrid high-rise scenarios**	■ 260 Parameters ● 2 Daylight metrics ▲ 20 ANN models	TEST 4
		■ 220 Parameters ● 2 Daylight metrics ▲ 20 ANN models	TEST 5
J5 **RQ5**	PART B **Optimising high-rise scenarios using the predictive models with Optimus and validation of the MUZO methodology**	■ 260 Parameters ● 20 Predictive models ▲ 3 Optimisation algorithms	TEST 4
		■ 220 Parameters ● 20 Predictive models ▲ 3 Optimisation algorithms	TEST 5
	CASE STUDY		
J6 **RQ6**	**Optimising Europoint complex for self-sufficiency in energy consumption and food production using MUZO and Optimus**	■ 117 Parameters ● 1 Self-sufficiency in energy 1 Self-sufficiency in food 1 Daylight metric ▲ 45 ANN models 13 Optimisation algorithms	
	CONCLUSIONS		

J Journal
RQ Research question

■ Parameter ● Performance ▲ AI Method

TOC

5 Case study

Optimising Europoint complex for self-sufficiency in energy consumption and food production in Rotterdam

Chapter 5 has been published as: Ekici, B.; Turkcan, O. F. S. F.; Turrin, M.; Sariyildiz, I. S.; Tasgetiren, M. F., Optimising High-Rise Buildings for Self-Sufficiency in Energy Consumption and Food Production Using Artificial Intelligence: Case of Europoint Complex in Rotterdam. Energies 2022, 15, (2), 660. For consistency of the dissertation, the layout is adapted to fit the template, some typos are adjusted, and phrases are reworded without changing the content.

https://doi.org/10.3390/en15020660

This chapter investigates utilising the MUZO methodology and Optimus tool to optimise the Europoint complex in Rotterdam, the Netherlands, for self-sufficiency in terms of energy consumption and food production. The sufficiency in food production is demonstrated for lettuce crops grown in vertical farms. Building-integrated photovoltaic panels are used in several building parts regarding sufficiency in energy. The optimisation problem, which involves 117 decision variables related to the façade design, and the thermal properties of the glazing, addresses the self-sufficiency at the building scale in detail. Moreover, another optimisation problem reports the potentials at the neighbourhood scale using the same self-sufficiency aspects and design parameters. Among 13 algorithms used to optimise both problems, the Optimus tool presented the most favourable self-sufficiency performance.

Optimising High-Rise Buildings for Self-Sufficiency in Energy Consumption and Food Production Using Artificial Intelligence

Case of Europoint Complex in Rotterdam

ABSTRACT

The increase in global population, which negatively affects energy consumption, CO_2 emissions, and arable land, necessitates designing sustainable habitation alternatives. Self-sufficient high-rise buildings, which integrate (electricity) generation and efficient usage of resources with dense habitation, can be a sustainable solution for future urbanisation. This paper focuses on transforming Europoint Towers in Rotterdam into self-sufficient buildings considering energy consumption and food production (lettuce crops) using artificial intelligence. Design parameters consist of the number of farming floors, shape, and the properties of the proposed façade skin that includes shading devices. Nine thousand samples are collected from various floor levels to predict self-sufficiency criteria using artificial neural networks (ANN). Optimisation problems with 117 decision variables are formulated using 45 ANN models that have very high prediction accuracies. 13 optimisation algorithms are used for an in-detail investigation of self-sufficiency at the building scale, and potential sufficiency at the neighbourhood scale. Results indicate that 100% and 43.7% self-sufficiencies could be reached for lettuce crops and electricity, respectively, for three buildings with 1800 residents. At the neighbourhood scale, lettuce production could be sufficient for 27,000 people, with a decrease of self-sufficiency in terms of energy use of up to 11.6%. Consequently, this paper discusses the potentials and the improvements for self-sufficient high-rise buildings.

KEYWORDS

Self-sufficiency, vertical farming, energy consumption, BIPV, building performance simulation, metropolis, artificial intelligence, machine learning, computational optimisation

5.1 Introduction

There is an increasing demand for the construction of high-rise buildings in metropolises [1], which requires the integration of various self-sufficiency aspects (such as food, energy, and water), owing to population growth and urbanisation trends. Compared with low-rise buildings, high-rises require more energy while causing significant CO_2 emissions [2]. For this reason, optimisation algorithms and machine learning (ML) techniques have been widely used for investigating sustainable high-rise alternatives. However, population growth does not only affect the increase in final energy consumption. Another global problem is the decreasing stock of arable land in the world [3]. The United Nations Food and Agriculture Organisation (FAO) foresees that only one-third of the arable land per person in 1970 will be available in 2050 [4]. Vertical farms, which are multi-storey plant factories [5], are designed to provide rapid and uniform product growth of high quality [6]. Recent work shows that various types of crops, e.g., leafy greens, lettuce, vine crops, and tomatoes [7], can be grown in closed farming systems. Therefore, integrating vertical farms in high-rise buildings can be one of the sustainable solutions for providing food in highly dense urban areas that can increase the amount of agricultural land while decreasing CO_2 emissions from the transportation of agricultural products.

Considering dense habitation, food production and energy generation in one building suggests a new optimisation problem in the architectural design called "*self-sufficient high-rise buildings*" (also defined as generative high-rises [8]). In this definition, self-sufficiency, which aims to provide a sufficient amount of resources in multiple aspects for the residents and for the neighbourhood in metropolitan areas, is different from self-sufficiency in terms of only energy (e.g., net-zero energy buildings (nZEB), energy-autonomous buildings [9]). Additionally, it is unlikely that high-rise buildings will be designed as off-grid systems with existing technology, such as small-scaled autonomous houses [10], because of their extreme sizes. Therefore, the contribution of self-sufficient high-rises is to generate and efficiently use multiple resources (such as energy, food, and water) to decrease their environmental impact while providing dense habitation in metropolitan areas. The complexity of this design problem is higher than the ones focusing on optimising high-rise buildings for various performance aspects of sustainability because of the existing and proposed challenges listed below:

TOC

- Providing a sufficient amount of food for at least the residents of high-rises and for as much of the neighbourhood as possible (proposed).

- Generating energy via solar power for food production and for the annual usage of the residents of the building (proposed).

- Integrating multiple performance aspects such as energy consumption, comfort and daylight (existing) [11].

- Considering performance variations between the ground and sky levels because of the dense urban areas in metropolises (existing) [12].

- Discovering well-performing high-rise alternatives in a reasonable amount of time during the conceptual design phase (existing) [13].

- Coping with the enormous number of decision variables to optimise the entire shape of high-rise buildings (existing) [14].

This paper investigates the optimisation of high-rise buildings for self-sufficiency in food production and energy consumption subject to daylight availability. The food production system involves stacked lettuce crops, which is only one of the possible agricultural products among the other alternatives mentioned before. Energy consumption considers the farming and residential energy usage and the generated energy via solar panels. While investigating high-rise alternatives, sufficient natural lighting is also taken into account. Europoint complex in Rotterdam (also known as Marconi Towers), designed by SOM architecture firm and constructed in 1975, is the focus of the work. The studied model initially focuses on the building scale, and further investigation addresses the potential self-sufficiency at the neighbourhood scale for sustainable future cities.

5.1.1 Problem statement

Optimising high-rise buildings for various design and performance aspects have been focussed on for two decades. Because of the existing challenges mentioned in the previous section, most of the published studies focus on the efficient usage of resources in high-rises, i.e. energy-efficient layout plans [15], natural ventilation potentials [16], energy-saving solutions for the envelope design [17], optimum solar access in high-density urban areas [18], double-skin façades for efficient

TOC

energy usage [19], optimisation processes for improving thermal and power performances [20], multiple building operation scenarios [21], and passive design strategies [22]. Considering the existing and proposed challenges together can result in a conflict between self-sufficiency aspects that increases the complexity of the high-rise design problem. For instance, closed farming systems consume significantly more energy in comparison to residential or office buildings, reaching more than 1000 kWh m^{-2} y^{-1} depending on the climate zone [23]. They do this while causing an increase in energy consumption with less habitation in the high-rise buildings; the higher number of farming floors provides more food with low CO_2 emissions when compared to regular farming. On the other hand, fewer farming floors provide less cultivation area with reduced energy consumption, but this causes higher CO_2 emissions owing to the transportation of the crops to respond to the food demand in metropolises. Therefore, an optimal solution for self-sufficiency in energy and food should be investigated. The multi-zone optimisation (MUZO) methodology, which uses artificial intelligence (AI) methods, is adopted in this research to cope with existing and proposed challenges [13,14]. Similar works, which focus on self-sufficiency in high-rises, are mentioned in Section 5.1.2, and the novelty of this paper is explained in Section 5.1.3.

5.1.2 Overview of previous works

Previous works were found using the keywords related to "high-rise building", "self-sufficiency", "energy consumption", "photovoltaic panel", "vertical farming", "daylight", and "artificial intelligence". Although a remarkable number of papers are published on high-rise optimisation for performance aspects related to sustainability, studies optimising self-sufficiency aspects are limited (Table 5.1). An early study focussed on optimising various floor plan configurations using the HGPSPSO algorithm to minimise the overall energy consumption considering building integrated photovoltaic (BIPV) panels [24]. Another work utilised single-objective and multi-objective optimisation algorithms for a two-step optimisation process using NSGA-II and HGPSPSO to minimise energy consumption while maximising energy production through opaque photovoltaic (PV) panels on the façade and semi-transparent PV panels as glazing [25]. A similar problem formulation was used to minimise energy demand and maximise the percentage of total comfortable time for achieving zero-energy high-rise buildings with NSGA-II [26]. In addition to producing energy through BIPV panels, a recent study considered the economic aspects of various hybrid renewable energy generation systems in high-rises collecting the results from four different applications [27]. Four of the previous studies presented promising results but only considered self-sufficiency in energy. The complexity of the studied design problems was limited because of the low size of decision variables (DV). Only one study focussed on the impact of the urban context [24], while the other papers only examined the high-rise building itself. Moreover, none of the previous studies investigated the results conducted by different optimisation algorithms that can lead to a representation of the solutions in local minima owing to the No Free Lunch (NFL) theorem [28]. Finally, ML algorithms were not involved in the previous studies. Hence, a limited number of function evaluations (FES) were considered because of the simulation-based optimisation processes.

TABLE 5.1 Overview of previous works

Study	Location	Aspects	Building Sufficiency	Neighbourhood Sufficiency	System	Optimisation Method(s)	DV Size
This paper	*Rotterdam*	*Food* *Energy*	*100%* *for 1800 people* *43.7%*	*100% up* *to 27000 people* *Between* *47.8%–11.6%*	*Stacked lettuce* *BIPV*	*given in Table 5.2*	*117*
[24]	Hong Kong	Energy	up to 48.77%	-	BIPV	HGPSPSO	11
[25]	Hong Kong	Energy	up to 71.36%	-	BIPV	HGPSPSO NSGA-II	11
[26]	Athens	Energy	33%	-	BIPV	NSGA-II	8
[27]	Hong Kong	Energy	16.02% 53.65% 69.26% 81.29%	- - - -	BIPV BIPV-wind BIPV-wind-battery Optimum BIPV-wind-battery	- - NSGA-II NSGA-II	- - 1 2

5.1.3 Novelty of this paper

This study considers multiple self-sufficiency aspects (i.e., food production and energy consumption) instead of considering a singular criterion (e.g., energy sufficient only as in previous works). Additionally, an enormous number of design parameters related to the number of farming floors, shape, and the property of the proposed façade skin with shading devices are extensively investigated using AI methods for self-sufficiency in building and the neighbourhood scales. For these reasons, the paper is focussed on optimising three towers of the Europoint complex in Rotterdam for self-sufficiency in total energy consumption (E_{tot}) and food production (F_P).

Since the developed model can consider the abovementioned crops by changing the simulation parameters to provide the required indoor environments, self-sufficiency for food production is demonstrated for stacked lettuce crops. The study aims to provide sufficient lettuce crops for the habitants with low energy consumption subject to acceptable daylight performance. As the urban context may affect the design decisions [12], the impact of two towers on another is considered by dividing each building into three subdivisions, as suggested in MUZO. Therefore, design decisions for various floor levels, which can provide better high-rise performance [29], are also investigated considering dense surroundings. A new façade skin is proposed in each tower for integrating PV panels and generating shading devices. 39-variables are used to parametrise the studied complex for the abovementioned design parameters. One optimisation problem is formulated for the three towers to use the advantage of each building's location for power generation so that the assignment of farming floors achieves the highest self-sufficiency performance possible. This formulation suggests a design problem, which corresponds to more than 4.5e + 91 design alternatives in the search space, with 117-variables.

A parametric high-rise model is integrated into the simulation engines to evaluate the self-sufficiency of the complex. Because the simulation models take significant time during the optimisation process, 45 surrogate models are developed for performance prediction based on feed-forward neural networks (FNN). Ten-fold cross-validation (CV) and hyperparameter tuning in each model are considered to investigate the highest prediction accuracies. Developed surrogate models are optimised for self-sufficiency in building and the neighbourhood scales using five single-objective and eight multi-objective optimisation algorithms. Near feasibility threshold (NFT) constraint handling [30,31] is used to cope with 37 and 36 constraints in both optimisation problems. Considering different search strategies suggests a need for a deep investigation of the search space since the global optimal of the design problem is unexplored owing to the NFL theorem in architecture [14]. Employed algorithms, based on the swarm, evolutionary and model-based search strategies that are frequently used in the architectural design domain [32], are given in Table 5.2.

TOC

TABLE 5.2 Overview of the optimisation algorithms used in this study

Scale	Objective	Constraints	Plug-Ins	Algorithms
Building	Minimise (E_{tot})	(F_P) 36 DF	Optimus (v1.0.2) [33] Silvereye (v1.1.0) [34] Galapagos (Rhino 6) [35] Opossum (v2.2.4) [36] “	jEDE [33] PSO [37] GA [38] CMA-ES [39] RBFopt [40]
Neighbourhood	Minimise (E_{tot}) Maximise (F_P)	36 DF	Optimus (v1.0.2) [33] Wallacei (v2.65) [41] Octopus (v0.4) [42] “ Opossum (v2.2.4) [36] “ “ “	jEDE (stepwise) [33] NSGA-II [43] HypE [44] SPEA-2 (Alt Pm/Pm) [45] RBFMopt [40,46] NSPSO [47], MACO [48] MOEA/D [49]

5.2 Methodology

The MUZO methodology [13,14], which consists of three main phases to optimise high-rise buildings for performance aspects related to sustainability, is considered to be the core of the methodology. The parametric high-rise model, alongside machine learning for surrogate models, computational optimisation and decision-making phases are followed to utilise the MUZO methodology to optimise the Europoint complex for self-sufficiency in food and energy as illustrated in Fig. 5.1. The following subsections explain the case-building and utilisation of the MUZO methodology in detail.

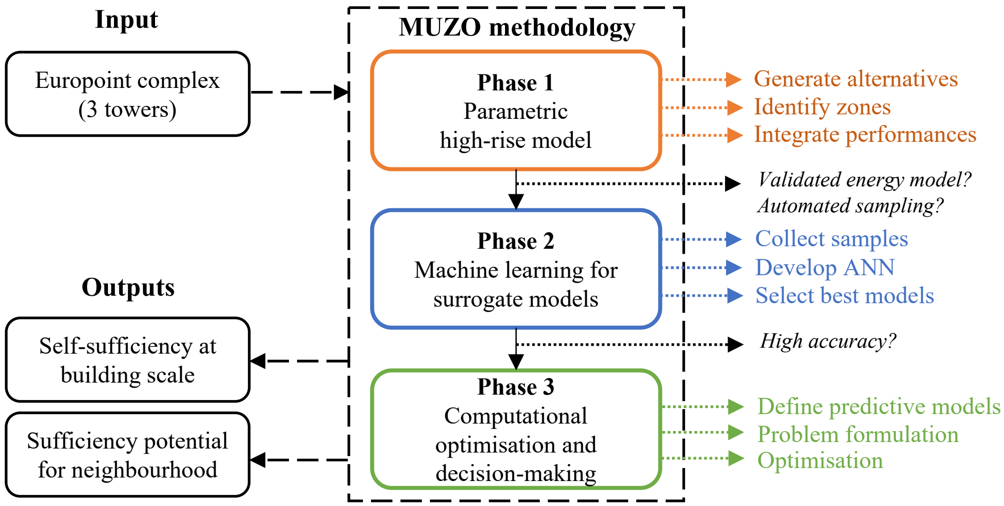


FIG. 5.1 Utilisation of MUZO methodology for the self-sufficient Europoint complex

5.2.1 Case building description

The Europoint complex, which was designed by the SOM architecture firm and constructed in 1975, is in the Merwe-Vierhavens (M4H) area of Rotterdam. The complex consists of three towers with 22 storeys, 3.75 m floor heights and a 95 m overall height. It has a rectangular plan scheme measuring 47.6 m by 33.2 m and a central core plan with 1580 m^2 gross and 1033 m^2 net floor areas; the Europoint Towers are some of the buildings that represent the international façade style of the 1970s (Fig. 5.2). Recently, the MOR team of TU Delft [50] proposed a prototype for the Europoint complex considering net-positivity in energy, air, water, material, and biomass for the Solar Decathlon competition in 2019 [51]. One of the main reasons to focus on these towers was the undesirable energy performance of the existing buildings, which corresponds to 75% of the building stock [52]. The Europoint complex is one of the many examples available in Europe of building complexes that have a significantly higher energy usage when compared to buildings that incorporate the sustainable solutions of the 21st century. In addition to the great potential for improving the energy performance of the Europoint complex, another aspect we consider in this study is to cope with the dense surrounding that will be a challenge in many metropolises based on population growth and urbanisation trends in the future. Besides the built environment, the distances between the Europoint Towers (31 m and 24 m, respectively) cause a large degree of shading on one another.

Therefore, each tower should be focussed on as a design problem that uses different parameter sets. Related to the self-sufficient high-rise concept, we also propose a different building program that provides farming and residential floors, in addition to public and commercial usage with semiprivate gardens, as illustrated in Fig. 5.3.

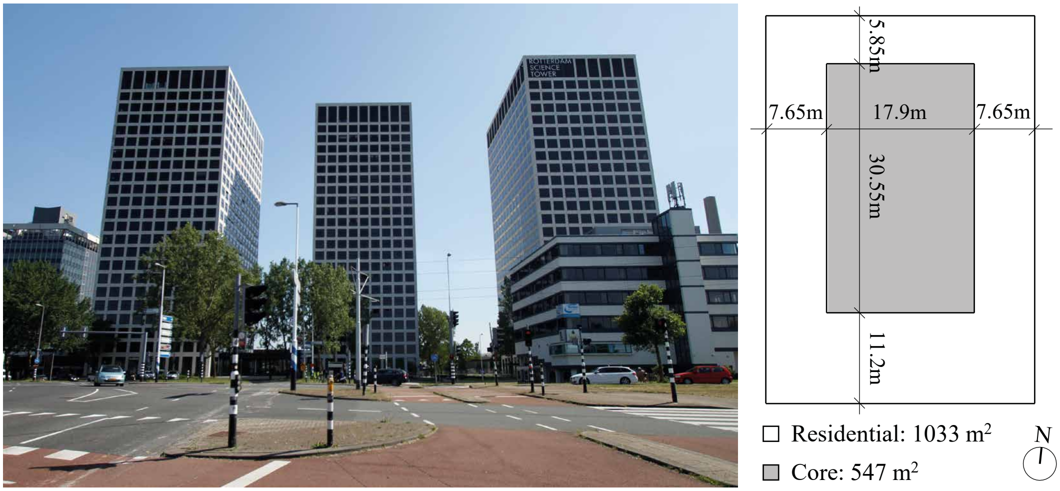


FIG. 5.2 Europoint complex in Rotterdam, the Netherlands

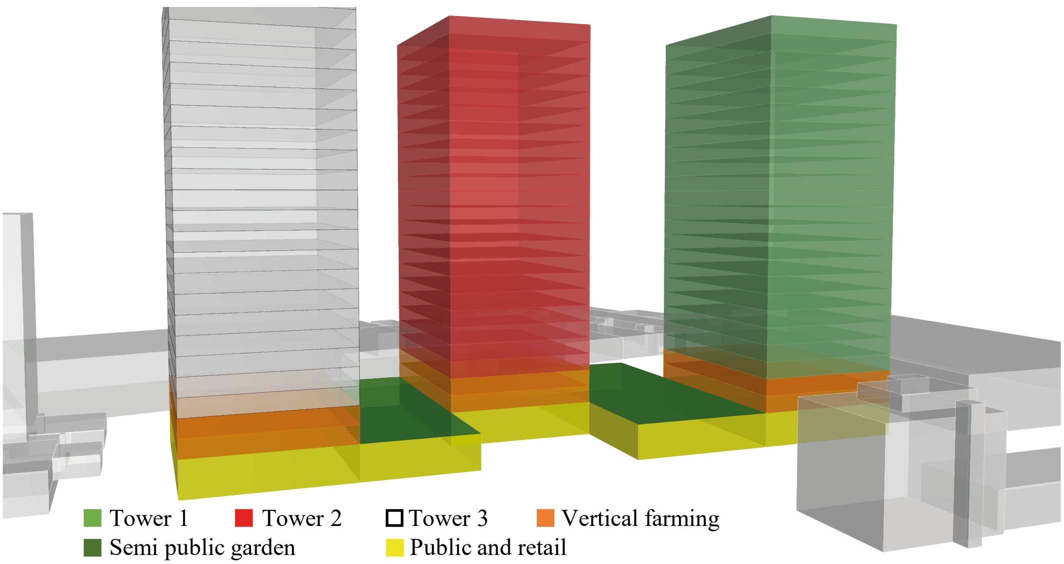


FIG. 5.3 Proposed building program

5.2.2 Parametric high-rise model

The studied buildings are parametrised considering the most commonly used design parameters for optimisation problems in architecture [32], as well as by following the steps of the first phase of MUZO methodology, which suggests dividing the buildings into several zones (subdivisions) to evaluate their performances separately. Hence, the performance variances in different floor levels can be considered during the optimisation process. The existing complex with surroundings is modelled in the Rhino3d computer-aided design program [53]. The parametrisation process is completed in the Grasshopper3d (GH) algorithmic modelling environment [54] that works as a plug-in for Rhino3d. The following subsections explain the parametrisation process and the simulation setups.

5.2.2.1 Parametrisation process

Public and commercial activities are placed on the ground level. Floor levels between 1 to 10, which are also decision variables, are associated with farming floors. The rest of the floor levels are defined as residential floors. The zoning process of the MUZO methodology is considered for dividing the residential part into three zones (subdivisions) in each building. The proposed façade covers the south (S), east (E), and west (W) orientations since the north (N) part of the buildings has insufficient solar potential and creates unnecessary shading. The shape of the skin also follows the zones, farming floors, and other parameters related to the BIPV. The proposed skin can have different distances from the building to create box-shaped shading devices in each zone. Reflectance values of these devices are defined as another parameter for extensive daylight control. In addition, 13 different glazing types, which are based on various visible transmittances (Tvis), thermal transmittances (U-val.) and solar transmittances (g-val.), are also investigated during the optimisation process. Finally, another parameter, which is used to control the existing size (2.7 m) of the windows, affects the daylight performance, energy consumption, and façade area that can be used for placing additional BIPV panels. Hence, the energy performance of the entire complex can be optimised for finding the most desirable self-sufficiency score under various circumstances (such as considering different design preferences for shaded or unshaded parts of the towers). All the mentioned variables, which are given in Table 5.3, correspond to a 39-dimensional design problem in each tower that suggests a 117-dimensional design problem for the entire complex. The proposed façade skin with three zones is illustrated in Fig. 5.4.

TABLE 5.3 Decision variables and glazing properties

Parameters	Explanation	Tower #			Zone #			Location	Type	Unit	Boundary
		1	2	3	1	2	3				
x_1	Number of farming floors	✓	✓	✓	-			-	Discrete	-	[0,10]
x_2, x_3	Extrusion of farming BIPV	✓			-			-	Discrete	m	[0,25]
x_2, x_3	“		✓		-			-	Discrete	m	[0,10]
x_2, x_3	“			✓	-			-	Discrete	m	[0,20]
x_4, x_5	Extrusion of roof BIPV	✓	✓	✓	-			-	Discrete	m	[0,5]
$x_6, \dots, x_9$	Glazing type	✓	✓	✓	✓	✓	✓	N-S-E-W	Discrete	-	[1,13]
$x_{10}, \dots, x_{12}$	Shading reflectance	✓	✓	✓	✓			S-E-W	Discrete	-	[0.3, 0.6, 0.9]
$x_{20}, \dots, x_{22}$	“	✓	✓	✓		✓		S-E-W	Discrete	-	[0.3, 0.6, 0.9]
$x_{30}, \dots, x_{32}$	“	✓	✓	✓			✓	S-E-W	Discrete	-	[0.3, 0.6, 0.9]
$x_{13}, \dots, x_{15}$	Shading distance	✓	✓	✓	✓			S-E-W	Discrete	m	[0.25, 1.50]
$x_{23}, \dots, x_{25}$	“	✓	✓	✓		✓		S-E-W	Discrete	m	[0.25, 1.50]
$x_{33}, \dots, x_{35}$	“	✓	✓	✓			✓	S-E-W	Discrete	m	[0.25, 1.50]
$x_{16}, \dots, x_{19}$	Window reduction size	✓	✓	✓	✓			N-S-E-W	Continues	m	[0.0, 1.0]
$x_{26}, \dots, x_{29}$	“	✓	✓	✓		✓		N-S-E-W	Continues	m	[0.0, 1.0]
$x_{36}, \dots, x_{39}$	“	✓	✓	✓			✓	N-S-E-W	Continues	m	[0.0, 1.0]

Glazing Types	Configuration	Argon	Air	Krypton	Type	Tvis	g-val.	U-val.
1	4–16–4	✓			Double	0.8	0.75	2.6
2	4–12–4		✓		Double	0.79	0.55	1.6
3	4–16–4		✓		Double	0.79	0.55	1.3
4	4–16–4	✓			Double	0.71	0.44	1.1
5	5–15–12	✓			Double	0.78	0.63	1.1
6	5–10–4		✓		Double	0.7	0.49	0.8
7	4–12–4–12–4			✓	Triple	0.7	0.6	0.7
8	9–10–4–10–13			✓	Triple	0.64	0.35	0.5
9	4–16–4–16–4	✓			Triple	0.69	0.48	0.6
10	4–12–4–12–4		✓		Triple	0.63	0.39	0.9
11	6–12–5–12–12			✓	Triple	0.62	0.42	0.4
12	6–12–4–12–8			✓	Triple	0.72	0.51	0.7
13	4–15–4–15–4	✓			Triple	0.7	0.74	0.6

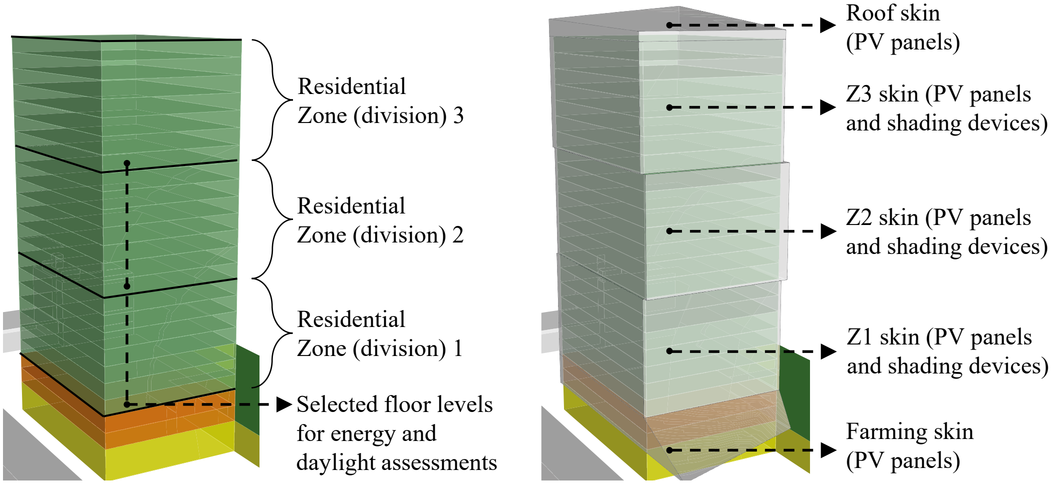


FIG. 5.4 Proposed façade design and building zones (subdivisions)

5.2.2.2 Simulation setups

The E_{tot} simulated for self-sufficiency of residential and farming floor levels consists of heating (E_h), cooling (E_c), lighting (E_L) and equipment (E_{eq}) consumption, the DF of residential levels and energy generation (E_g) via BIPV panels. Honeybee (HB) and Ladybug (LB) plug-ins in GH [55], which use Open Studio for energy analysis [54], simulating the energy consumption and generation. Regarding the daylight assessment, HB and LB also evaluate the design alternatives using the Radiance simulation engine [56]. Having an oceanic climate, winters in Rotterdam are mild, humid, and windy, whereas summer days are cool. Therefore, the focus of residential energy consumption is primarily to minimise the heating, lighting and equipment loads. In the farming system, significant energy usage is necessary for the growing process of the plants that require cyclically consistent temperatures, a certain level of humidity with mechanical ventilation, artificial lighting, and other mechanical equipment. Therefore, heating and cooling loads are both considered, as well as lighting and equipment loads. The simulation model of the residential floors is simplified to five thermal zones as N, S, E, W, and core, whereas four thermal zones are utilised in the farming floors, which are farming, germination and seed (g&s), and core, as illustrated in Fig. 5.5. Inputs of the farming energy model are based on recently published closed systems [7,23]. Schedules are based on a 16/8 occupancy period, and 1000 ppm CO_2 is considered, which results in an estimated 80 kg m^2 y^{-1} lettuce yield in an 833 m^2 floor area, for three stacked hydroponics crop systems covering 50% of the floor plan. In residential floors, various schedules are considered for occupancy, lighting, equipment, and HVAC, considering the preferences of the MOR team [50] as defined in Fig. 5.6. All the other inputs of the energy models are given in Table 5.4.

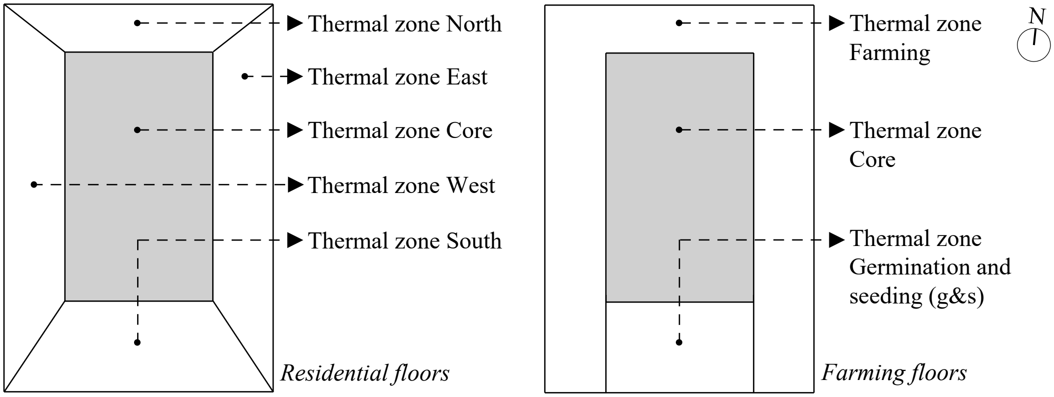


FIG. 5.5 Thermal zones of energy model in residential and farming floors

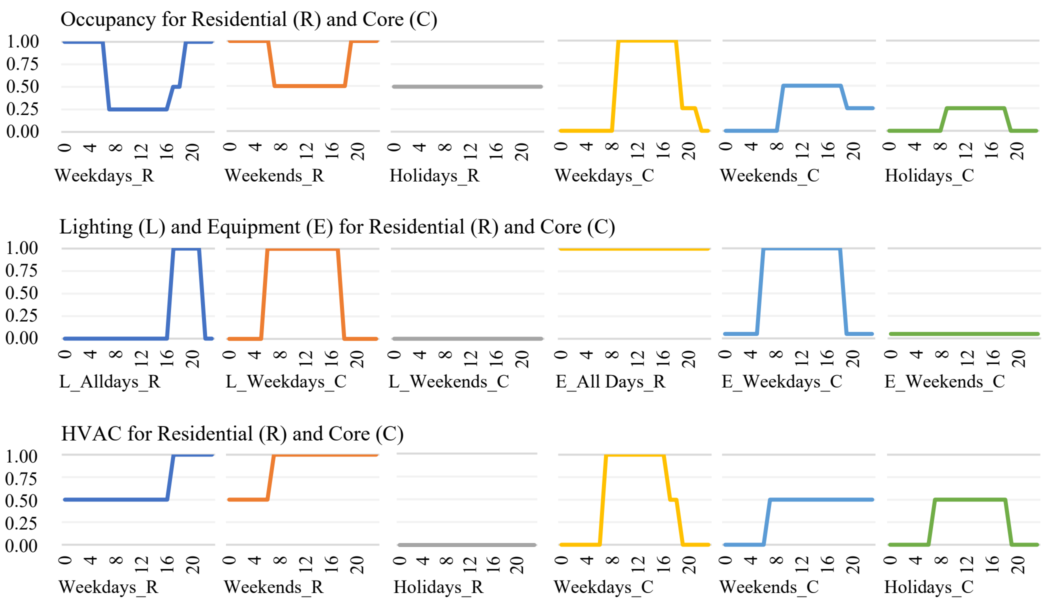


FIG. 5.6 Schedules used in the residential energy model

Regarding BIPVs, the rooftop and façade of residential floors are used to place solar panels. Additionally, external surfaces of the farming floors are used to locate BIPVs since the exterior surfaces of the considered farming system consist of opaque wall material. On the southern elevation of farming, and on the rooftop, four different parameters are defined to optimise the alignment of the panels to achieve the highest E_g.

The boundaries of these parameters on the street level (x_2, x_3) are defined according to site conditions. Therefore, different maximum extrusions are considered for each building (Table 5.3). The radiation analysis to calculate the energy potential of BIPV surfaces is conducted on a 0.5 m by 0.5 m grid to calculate the energy potential through radiation analysis. Sensor points with a minimum of 175 kWh m^{-2} of energy falling on the surface are used to locate the PV panels for maximum energy/cost profit. This selection also suggests different BIPV patterns in each tower because the way each building is shaded differs from one another. Hence, the total E_g can be enhanced in the entire complex because of the larger available PV surface area in different subdivisions (zones) of the towers. As a result, an optimum configuration for window sizes, glazing types, and shading extrusion is expected for energy consumption, alongside generation, and daylight availability. Fig. 5.7 illustrates an example of BIPV allocation in the Europoint complex, whereas parameters used in the calculation of E_g are given in Table 5.5.

TABLE 5.4 Inputs of the energy models

Type	Simulation Parameter	Unit	Validation (Simulation)	Sampling (Simulation)	Residential Model	Farming Model
Loads	Equipment	W/m^2	Residential: 5.5	Residential: 5.5	✓	
		"	Core: 2.5	Core: 2.5/2.0	✓	✓
		"	-	Farming: 0		✓
		"	-	g&s: 0		✓
	Lighting	"	Residential: 1.5	Residential: 1.5	✓	
		"	Core: 7.5	Core: 7.5	✓	✓
		"	-	Farming: 100		✓
		"	-	g&s: 0/1.5		✓
	Mech. Vent.	$l/s\text{-}m^2$	0.9	0.9	✓	✓
	Natural vent.	-	Off	On	✓	
	Air-tightness	ac/h	0.1	0.1	✓	
			-	1.0		✓
	People	ppl/m^2	Residential: 0.04	Residential: 0.04	✓	
		"	Core: 0.08	Core: 0.08	✓	✓
		"	-	Farming: 0		✓
		"	-	g&s: 0.02		✓
Setpoints	Heating	°C	20/18	20/18	✓	
		"	-	24		✓
	Cooling	"	-	-	✓	
		"	-	30		✓
	Ventilation	"	-	21/24	✓	
	Humidity	%	Residential: 10/90	Residential: 10/90	✓	
		"	-	Farming: 75/85		✓
	Daylight	lx	Core: 300	Core: 300	✓	
		"	Residential: 250	Residential: 250	✓	
HVAC	Template	-	Ideal air loads	Ideal air loads	✓	✓
	Economiser	-	None	Differential Dry Bulb	✓	✓
	Heat recovery	-	Off	Off	✓	✓
	CoP	-	1.0	1.0/5.0	✓	✓
Construction type	Floor	-	Adiabatic	Adiabatic	✓	✓
	Ceiling	-	Adiabatic	Adiabatic	✓	✓
	Exterior wall	-	Generic	Metallic cladding/60 mm, wood framing/180 mm, cast concrete/560 mm	✓	✓
	Interior wall	-	Generic	Cast concrete/300 mm	✓	✓
	Glazing	-	#7 in Table 5.3	All types in Table 5.3	✓	

>>>

TOC

TABLE 5.4 Inputs of the energy models

Type	Simulation Parameter	Unit	Validation (Simulation)	Sampling (Simulation)	Residential Model	Farming Model
Construction U-val.	Floor	W/m² K	0.1538	0.22	✓	✓
	Ceiling	“	0.1538	0.22	✓	✓
	Exterior wall	“	0.1538	0.1538	✓	✓
	Interior wall	“	0.1538	0.40	✓	✓
	Glazing	“	#7 in Table 5.3	All types in Table 5.3	✓	

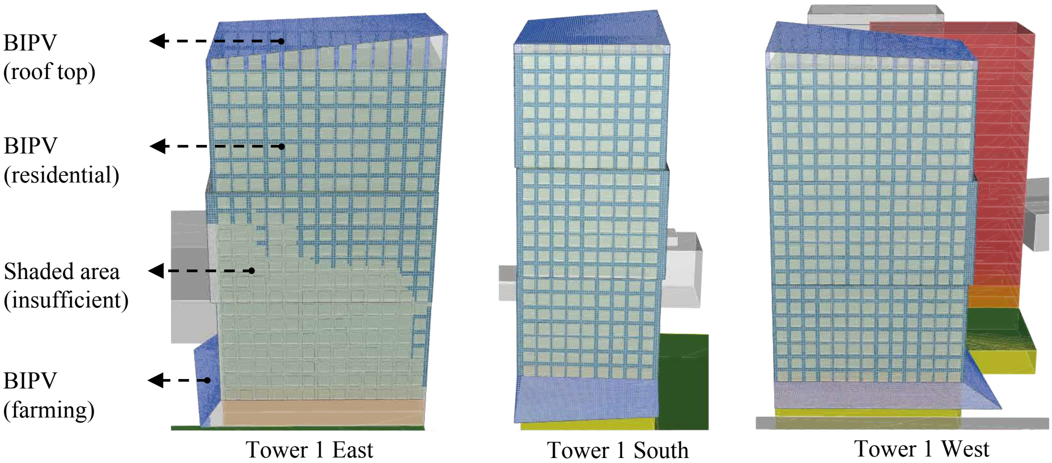


FIG. 5.7 BIPV allocation of Europoint complex

TABLE 5.5 Parameters of BIPV simulation

Location	PV Type	PV Efficiency	Coverage	Inverter Efficiency
Farming/roof façade	Opaque monocrystalline PV cells	20%	85%	90%
Rooftop	“	20%	85%	90%
Residential façade	Colorblast monocrystalline PV cells	14%	85%	90%

Finally, daylight models are developed to simulate DF, which is one of the most commonly used daylight metrics to identify the lighting performance in the early phase of the design process [57,58]. According to Dutch standard NEN-EN 17037, a minimum of 2% of DF should be provided in residential places. While evaluating the DF, performances of each orientation in each zone, which correspond to 36 DF results, are considered under an overcast sky. For each model, a 0.5 m by 0.5 m grid size, which is 0.8 m above the finished floor, is used. In total, 7263 sensor points are used to evaluate one design alternative for the entire complex that corresponds to 807 sensor points on one floor. An example simulation result with the sensor points is illustrated in Fig. 5.8.

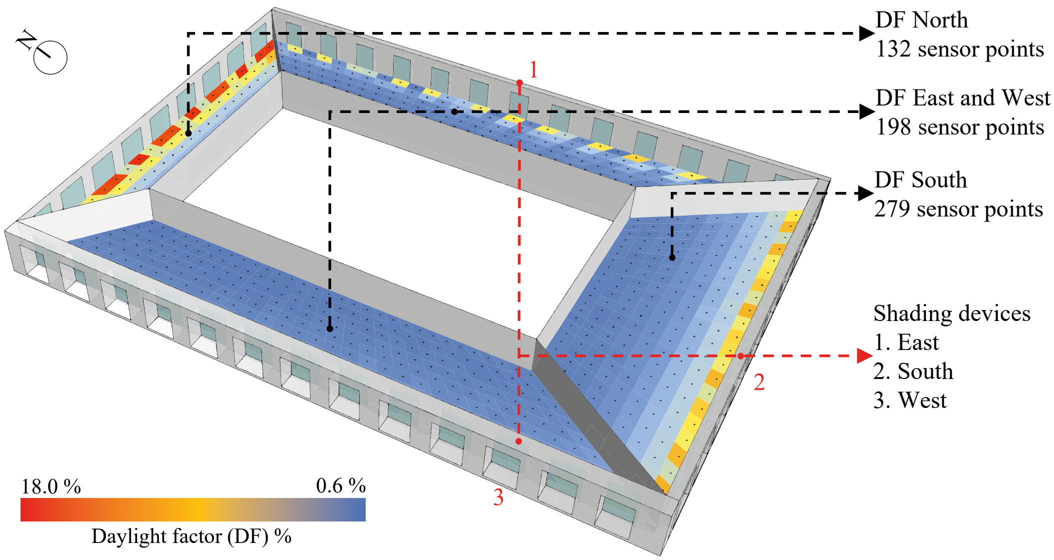


FIG. 5.8 Simulation result of *DF* for one design alternative in a floor level

Radiance parameters, which are similar to those used in the studies published in the same domain, and the material properties of the developed daylight models, are given in Table 5.6. Based on the simulation results in three zones for three towers, the following equations calculate the energy consumption of the entire complex:

TOC

$$E_{tot} = E_R + E_F - E_g \,, \tag{5.1}$$

$$E_R = \sum_{i=1}^{3}\sum_{j=1}^{3}\left(E_{h_{i,j}} + E_{L_{i,j}} + E_{eq_{i,j}}\right) z_{i,j} \,, \tag{5.2}$$

$$E_F = \sum_{i=1}^{3}\sum_{x_1=0}^{x_1} E_{h_{i,x_1}} + E_{c_{i,x_1}} + E_{L_{i,x_1}} + E_{eq_{i,x_1}} \,, \tag{5.3}$$

$$E_g = \sum_{i=1}^{3} E_{g_i} \,, \tag{5.4}$$

where E_R and E_F are the total energy consumption of residential and farming floor levels, $\{i_1,...,i_3\}$ are the three towers in the complex, $\{j_1,...,j_3\}$ are the three subdivisions (zones) of each tower, and $z_{i,j}$ is the number of each zone in each tower that changes with the number of the farming floors (x_1). Since the parameters related to the closed farming system are not considered in this study, simulation results of the farming model explained above are multiplied with x_1 to calculate the farming energy consumption in each tower. All the simulation models use the Amsterdam weather data file provided by LB tools [64]. Integrating simulation engines to the parametric model completes the first phase of the MUZO methodology. After validating the results of the energy model with simulations, the parametric high-rise model becomes ready for the second phase of MUZO.

5.2.3 Machine learning for surrogate models

The surrogate modelling in phase two of the MUZO methodology starts with the sampling process. In this paper, Latin Hypercube Sampling (LHS) [65] generates the design alternatives to be used in the ANN development. As a common approach, Equation (5.5) identifies the sample size as [66]:

$$n_s = 22.5 n_i \,, \tag{5.5}$$

where n_s is the size of the samples and n_i is the number of decision variables. In this case, at least 877 samples should be collected for a 39-dimensional design problem.

TABLE 5.6 Radiance parameters and material properties of the daylight simulations

Study	Ambient Accuracy (-aa)	Ambient Bounces (-ab)	Ambient Division (-ad)	Ambient Resolution (-ar)	Ambient Super-Samples (-as)
This paper	*0.1*	*4*	*1000*	*300*	*20*
[59]	0.15	2	1000	300	20
[60]	0.15	2	512	256	128
[61]	0.1	5	1500	300	20
[62]	0.15	2	512	256	128
[63]	0.15	2	512	256	128

Category	Type	Reflectance or Tvis	Tower #			Zone #		
			1	2	3	1	2	3
Exterior wall	Concrete	0.4	✓	✓	✓	✓	✓	✓
Interior wall	Painted white wall	0.7	✓	✓	✓	✓	✓	✓
Ceiling	Painted white ceiling	0.7	✓	✓	✓	✓	✓	✓
Floor	Wood	0.4	✓	✓	✓	✓	✓	✓
Shading device	White/grey/dark (see Table 5.3)	(see Table 5.3)	✓	✓	✓	✓	✓	✓
Glazing	(see Table 5.3)	(see Table 5.3)	✓	✓	✓	✓	✓	✓
Surrounding (city)	Concrete blocks	0.3	✓	✓	✓	✓	✓	✓
Surrounding (towers)	-	0.5	✓	✓	✓	✓	✓	✓
Ground	-	0.2	✓	✓	✓	✓	✓	✓

Since the extension of the sample size is beneficial [67], the size of the collection is extended to 1000; thus, the sampling process covers 9000 design alternatives to be collected from the entire complex for E_R, E_g, and DF. The normalisation of the collected samples initiates the development of the ANN models using the min–max scale as:

$$x' = \sigma\left(\max(x) - \min(x)\right) + \min(x), \tag{5.6}$$

where x' is the scaled value, σ is the standard deviation, and x is the original value. After identifying a ratio of 0.2 to define training and test sets, the Stochastic Gradient Descent (SGD) algorithm [68] optimises the weights and biases for an ANN architecture that has 39 input, 3 hidden, and 1 output layers. The α of each i^{th} layer is activated as in Equation (5.7), whereas each neuron is activated by the rectified linear units (ReLU) as in Equation (5.8):

TOC

$$\alpha_i = f\left(b_i + \sum_{j=1}^{m} w_{ij} x_i\right), \tag{5.7}$$

$$f(x) = \begin{cases} 0 & for \quad x \le 0 \\ x & for \quad x > 0 \end{cases}, \tag{5.8}$$

where f is the activation function, b is the bias, w_{ij} is the i^{th} layer of the j^{th} weight, and x_i is the input vector of the i^{th} layer. To avoid overfitting, ANN models also consider the dropout technique [69] with a rate of 0.1. Based on this setup, the grid search process initiates parameter tuning with a 10-fold CV to identify the best prediction accuracy in each ANN model using the five hyperparameters given in Table 5.7. The R-squared (R^2) term in Equation (5.9), the mean squared error (MSE) in Equation (5.10), the mean absolute error (MAE) in Equation (5.11), and the standard deviation (Std) in Equation (5.12) are used for model selection:

$$R^2 = 1 - \frac{\sum_i (x_i - y_i)^2}{\sum_i (x_i - \bar{x})^2}, \tag{5.9}$$

$$MSE = \frac{1}{n} \sum_{i=1}^{n} (x_i - y_i)^2, \tag{5.10}$$

$$MAE = \frac{\sum_{i=1}^{n} |y_i - x_i|}{n}, \tag{5.11}$$

$$Std = \sqrt{\frac{1}{n-1} \sum_{i=1}^{n} (x_i - \bar{x})^2}, \tag{5.12}$$

where n is the size of the samples, $\{x_1, \ldots, x_n\}$ are the observed values, $\bar{x}$ is the mean of the collected data, x_i is the observed data, and y_i is the predicted value. The purpose is to select the models which present a high mean for R^2, and a low mean for MSE, and MAE while presenting low Std values for R^2, MSE, and MAE at the end of the grid search. Hyperparameters with the best accuracies are once again fit to record the weights and biases for developing the predictive models in the last phase of the MUZO methodology. The abovementioned steps are developed in the Python programming language [70], which also uses the additional Python libraries given in Table 5.7 to automate the entire ML process.

TABLE 5.7 Python libraries and grid search setup

Python Libraries		Grid Search Setup	
Library	Explanation	Hyperparameters	Values
Pandas [71]	Data analysis library	Batch size	[25, 50, 75]
Keras [72]	Deep learning library	Epochs	[250, 500, 750]
TensorFlow [73]	Open-source ML platform	Neuron size	[50, 100, 150]
Scikit-learn [74]	ML library	Learning rate	[0.01, 0.05, 0.1]
Joypy [75]	Plot library	Momentum	[0.3, 0.6, 0.9]

5.2.4 Computational optimisation for decision-making

The final phase of the MUZO methodology initiates the optimisation process in GH, which reads the weights and biases that are recorded in phase two, by defining each predictive model with the following equation:

$$y = f_n(f_{n-1}(f_{n-2}(f_{n-3}(x \cdot w_{n-3} + b_{n-3}) \cdot w_{n-2} + b_{n-2}) \cdot w_{n-1} + b_{n-1}) \cdot w_n + b_n), \quad (5.13)$$

where f_n is the n^{th} activation function, w_n and b_n are the n^{th} weight and bias, respectively, and x and y are the input vector and the predicted performance aspect. The first optimisation problem focuses on the analysis of self-sufficiency in building scale in detail. Equation (5.14) presents the single-objective constrained problem formulation that is subject to 37 constraints for the first optimisation round:

$$\begin{aligned} \min: \quad & E_{tot} \\ subject \;\; to: \quad & DF_{1,\ldots,36} \geq 2, \\ & F_p \geq 60 \end{aligned} \quad (5.14)$$

where E_{tot} is the total energy consumption of the three towers, $DF_{1,\ldots,36}$ are the values of 36 DFs in four orientations (N, S, E, W) for the three zones of the three towers. The minimum F_p is defined as 60 tons for 1800 residents living in the Europoint complex, assuming 100 g of lettuce is consumed per person in one day. Therefore, at least two farming floors should be placed in the complex, while optimal energy production can also be achieved with the alignment of two farming floors in three buildings. The second optimisation round investigates the potential self-sufficiency of the Europoint complex at the neighbourhood scale with multi-objective constrained problem formulation subject to 36 constraints as:

$$\begin{array}{rl} \min: & E_{tot} \\ \max: & F_p \\ subject\ \ to: & DF_{1,\ldots,36} \geq 2 \end{array}, \tag{5.15}$$

Equation (5.15) considers F_p as another objective function that corresponds to 30 non-dominated solutions in the Pareto front. Instead of limited searchability for constant penalty functions, both optimisation problems handle the constraints with NFT by penalising the fitness function $f(x)$ as:

$$f_p(x) = f(x) + \left(\frac{v(x)}{NFT} \right)^{\alpha}, \tag{5.16}$$

$$NFT = \frac{NFT_0}{1 + \lambda.g}, \tag{5.17}$$

where $f_p(x)$ is the penalised fitness function, $v(x)$ is the total violation, NFT_0 is the upper bound of the NFT taken as 0.1, λ and α are defined as 0.04 and 2, respectively, and g is the iteration or generation number. The optimisation phase of the MUZO methodology suggests a need for replication of the optimisation runs and algorithm comparisons owing to the NFL theorem in architecture. Thus, the decision-making step considers various replications using different optimisation algorithms for extensively investigating the unexplored search space of the design problem.

5.3 Results and discussion

This section presents the results of the validation and sampling processes using simulation models, statistical results of the grid search and tuned models of ML, and the optimisation results for two scales of self-sufficiency problems, i.e., the building and neighbourhood scales. Finally, results are discussed for both problem scales focusing on the potentials and limitations of the study.

5.3.1 Model validation and sampling results

Energy results collected from HB were validated using another simulation model in DesignBuilder (DB) before initiating the sampling phase for the entire complex. Therefore, a simplified version of the residential energy model was used in HB and DB with the values given in Table 5.4. As illustrated in Fig. 5.9, monthly air, radiant, and operative temperatures suggested an observable correlation between the results of HB and DB. For further investigation, a regression analysis was performed considering weekly temperatures. The results in Fig. 5.10 indicated that R^2 for all the temperature results was higher than 0.96. Afterwards, E_F was calculated for the validated HB environment. When CoP was equal to 1, annual farming energy consumption was calculated as 1123.2 MWh y^{-1}, whereas it was 629.3 MWh y^{-1} for CoP 5. The F_p of the farming system for lettuce crops was calculated as 33 tons on a single floor with the temperatures given in Fig. 5.11.

The collected samples, which include the simulation results of E_R (for CoP 1 and 5), E_g and average DF for all orientations in all zones of the three towers, were published as an open-access dataset [76]. The completion time for this task was recorded as 15 d 6 h 40 m using a computer with an Intel I7 5820K core processor at 3.30 GHz, with 16-GB DDR4 of memory, and a 256-GB solid-state drive. Distributions of the simulation results (Fig. 5.12) indicated that the three towers presented similar E_R values between 1066 MWh and 1951 MWh (when CoP was equal to 5), and a wider range between 1652 MWh and 3246 MWh (while CoP was 1). Regarding E_g, tower one had the highest solar power potential with a range of between 592 MWh and 1040 MWh. While tower two presented a solar power potential range of between 580 MWh and 901 MWh, tower 3 demonstrated a range from 556 MWh to 923 MWh. Although ranges of towers two and three were similar, the minimum E_g of tower two was higher than that of tower three because of its location.

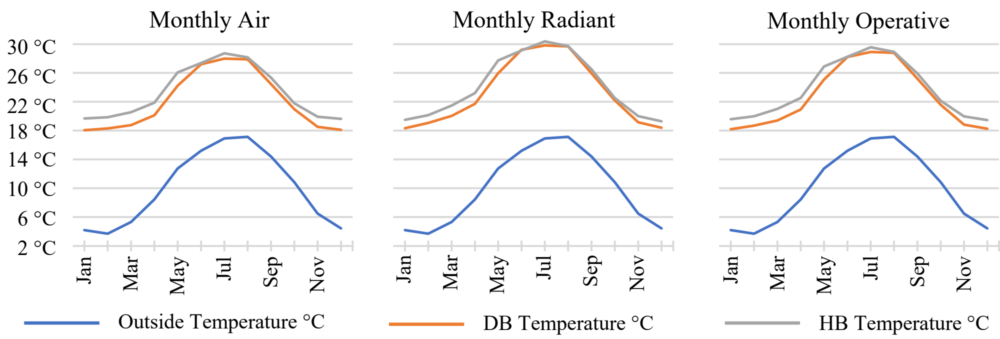


FIG. 5.9 Monthly temperature comparisons between DB and HB

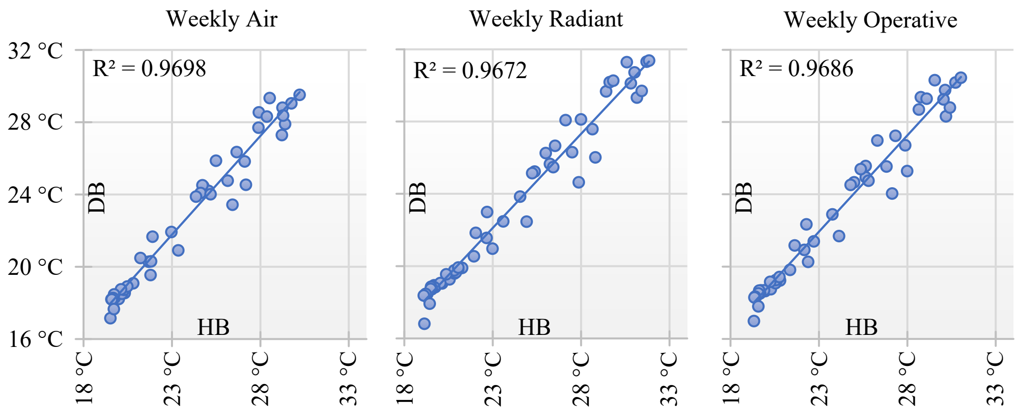


FIG. 5.10 Weekly temperature comparisons between DB and HB

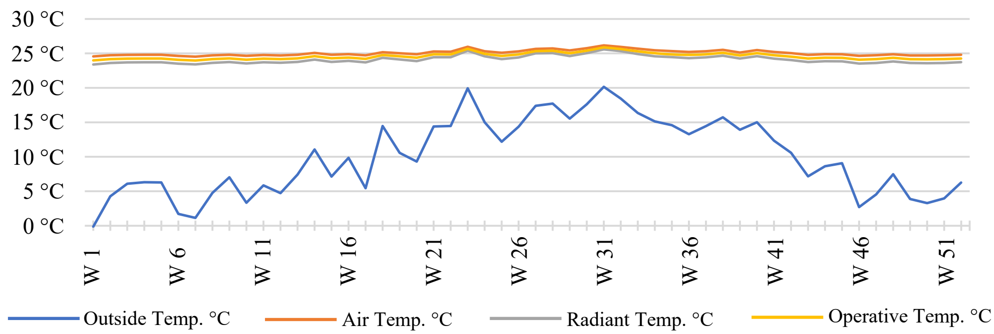


FIG. 5.11 Weekly (W) temperatures of the farming system

TOC

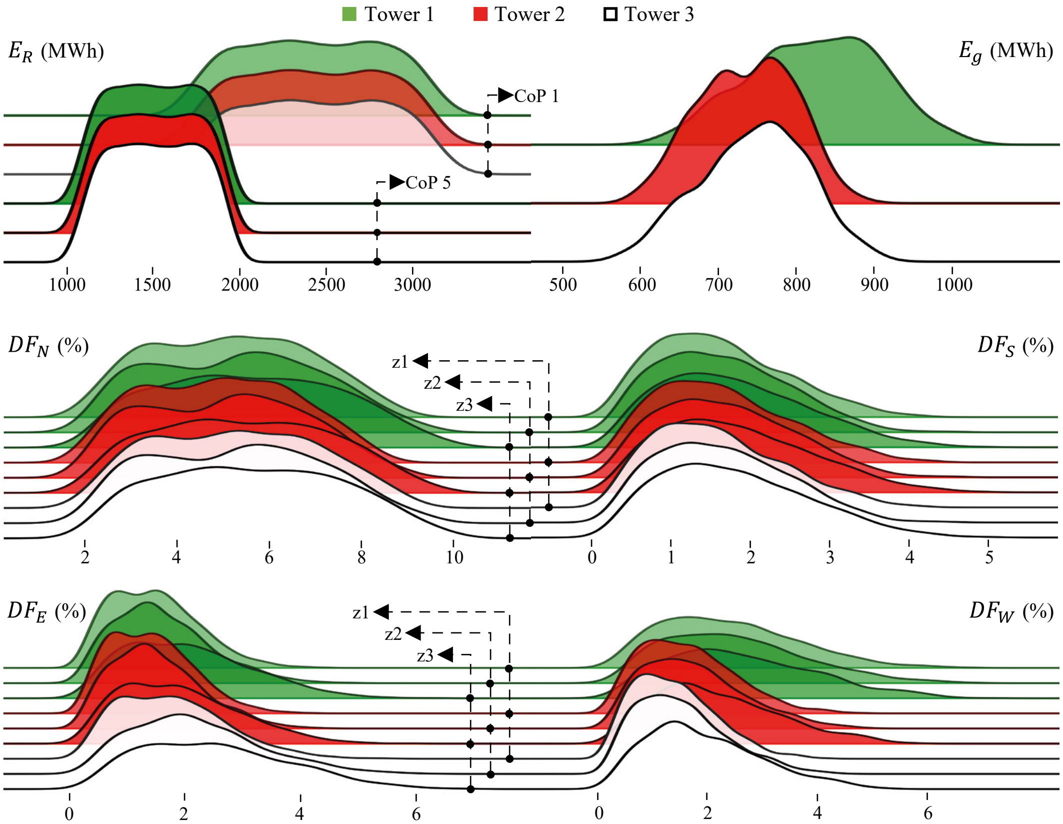


FIG. 5.12 Distributions of collected samples

Nevertheless, the maximum E_g of tower three indicated that its solar potential could be improved with the x_2 and x_3 parameters by controlling the positions of the BIPV panels. Regarding DF, N and S orientations presented similar distributions, while higher values were observed in the north when compared to the south. The reason for this was that the places located on the southern elevation of the towers were much deeper than in the north. On the other hand, various distributions were observed in the E and W orientations in different zones and towers, which could be due to the effect of the towers shading one another. Finding DF values higher than 2% in the E and W orientations was more challenging than for N and S orientations during the sampling process. Therefore, different design decisions were expected for various zones of each tower in the optimisation process.

5.3.2 Machine learning results

The data collection, which had 1000 samples in each zone, was trained using FNN (as an ANN method) and is explained in Section 5.2.3. In total, 9000 samples were used to develop 45 surrogate models, which predict DF, E_R, and E_g using 39 design parameters that are given in Table 5.3. Additionally, 36 DF models were developed separately for each zone in each orientation, since the daylight standard should be achieved for the entire complex. During the prediction of E_{tot}, the summation of consumed (E_R) and produced (E_g) energy for each tower was considered. Grid search setup, which investigates the five hyperparameters given in Table 5.7, was considered with a 10-fold CV. The total number of the models to fit using the FNNs was calculated as 109,350, based on the results of 2430 models for completing the grid search process in every fold of the 45 surrogate models. The average completion time for one grid search task was recorded as 2 h 17 m using a computer with an Intel Xeon E5-2640 v4 core processor at 2.40 GHz, with 64-GB DDR4 of memory, and a 1024-GB solid-state drive. A self-developed Python program automatically read the data, developed ANN models, selected the best hyperparameters for 45 models according to mean and Std of the MAE, MSE, and R^2 values, fitted the final ANN models for training alongside test sets using the selected hyperparameters, and reported the statistical results as well as weights and biases for the 45 surrogate models. The best hyperparameters with corresponding means and Stds of the MAE, MSE, and R^2 values for three zones of the three towers are given in Fig. 5.13.

During the batch size selection, 21 models had the best accuracy using a value of 25, while for ten models it was 50, and for 14 models it was 75. For epochs, 31 models had the best score with values of 750, for nine models it was 500, and for five models it was 250. For the momentum parameter, four models had the best score using values of 0.3, for seven models it was 0.6, and for 34 models it was 0.9. Regarding the learning rate parameter, nine models had the best accuracy using values of 0.01, fifteen models yielded the best results using 0.05, and 21 models using 0.1. As final the hyperparameter, 6 out of 45 ANN models had the best accuracy using 50 neurones, 23 models had the best results using 100 neurons, and for 16 models the best value was 150 neurones. Results of the best hyperparameters indicated that the mean of all MAE values was less than 0.05 while having Std values of less than 0.005. The mean of the MSE values was less than 0.003 with Std values smaller than 0.0005. The mean of all R^2 values was higher than 0.94, whereas the Std values were less than 0.0015.

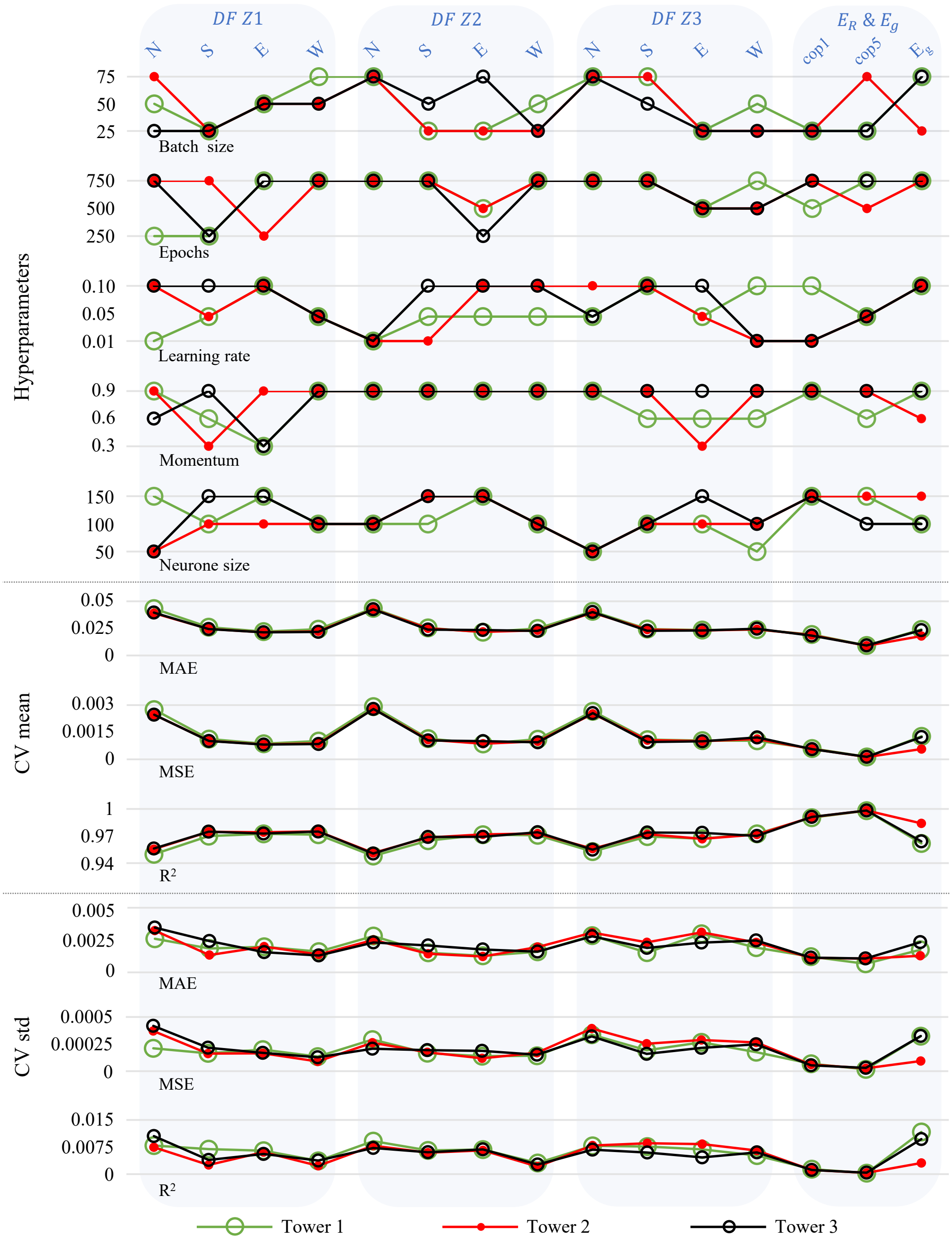


FIG. 5.13 Grid search results of the best ANN models

Reported statistical results in the grid search process presented promising prediction accuracies. Therefore, tuned ANN models were developed using the selected hyperparameters in the next step. Initially, the data was split into training and test sets considering a ratio of 0.2 to demonstrate accurate prediction. Results in Fig. 5.14 indicated that MAE and MSE values of both sets were less than 0.05 and 0.003, respectively. Additionally, all R^2 values were higher than 0.94. Finally, tuned ANN models were used to predict the parameter values generated by LHS with the weights and biases provided as the supplementary material. R^2 values indicated that there was a very high correlation between the simulation and the prediction results for E_R, E_g and mean DF as shown in Fig. 5.15. All of the results of the second phase of the MUZO methodology suggested that the predictive models could be used for the optimisation process in the next phase.

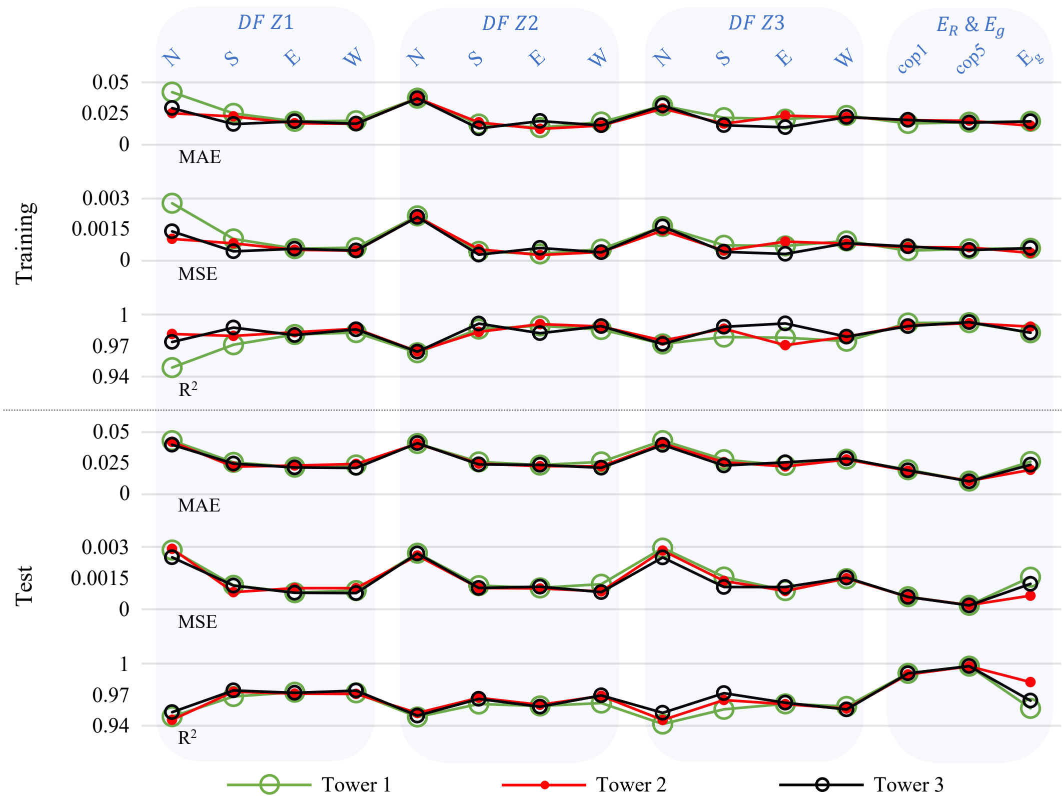


FIG. 5.14 MAE, MSE, and R^2 results of training and test sets

TOC

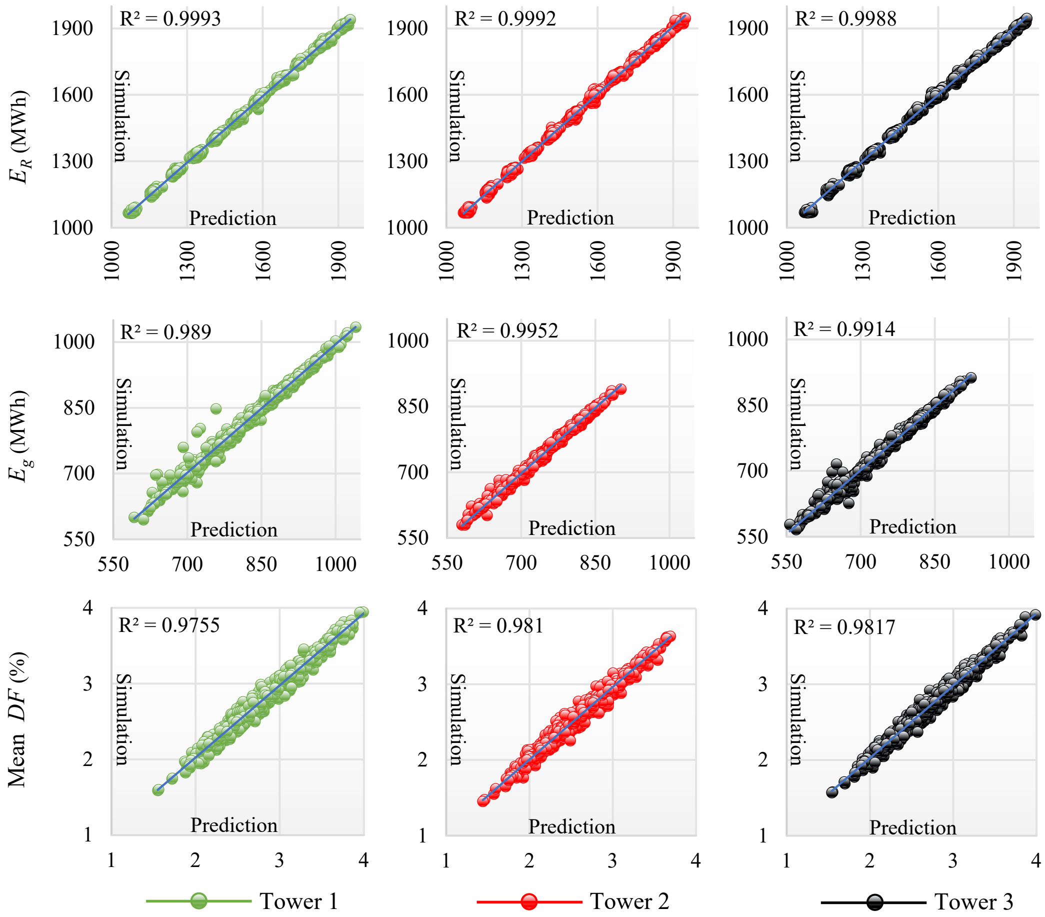


FIG. 5.15 Simulated results versus predicted results

5.3.3 Computational optimisation results

During the last phase of the MUZO methodology, two design problems were considered focusing on different scales. First, a single-objective problem was investigated in detail employing the single-objective optimisation algorithms given in Table 5.2, using the Optimus, Opossum, Galapagos, and Silvereye plug-ins in GH. Since jEDE, GA, and PSO are populated algorithms, 50 and 100 population & swarm (P&S) sizes were considered in this optimisation round. Secondly, the multi-objective optimisation problem was investigated considering single replication to present

TOC

the potential self-sufficiency of the Europoint complex at the neighbourhood scale by employing the stepwise and multi-objective optimisation algorithms given in Table 5.2, using the Optimus, Wallacei, Octopus, and Opossum plug-ins in GH. Except for RBFMOpt, 50 P&S was considered for populated algorithms. The mutation rate of jEDE was set between −1 and 1, while the other algorithms were used with the default parameters. The NFT module in the Optimus plug-in was considered to cope with the constraint functions of both problems for all optimisation algorithms. All the runs were completed when CoP was equal to 5. Since the ML models for CoP 1 were also developed, their results were also reported. The following subsections present the gathered results for both problems at the building and neighbourhood scales.

5.3.3.1 Building scale

Results at the building scale were conducted for 7500 FES using a computer with an Intel I7 5820 K core processor at 3.30 GHz, with 16-GB DDR4 of memory, and a 256-GB solid-state drive. Because the non-populated algorithms were used in the Opossum plug-in, 7500 FES was defined as a maximum iteration count for CMA-ES and RBFOpt, whereas 50 and 100 P&S sizes were considered with 150 and 75 generation/iteration counts for the populated algorithms. The criteria for comparing the optimisation results of this problem were defined as finding feasible solutions for 37 constraints, the lowest E_{tot} needed to have zero violation, Std was calculated for five replications, and computation time (*CPU*) needed to have been recorded. Fig. 5.16 illustrates the boxplots of the optimisation results for feasible solutions only, whereas Table 5.8 presents the number of feasible solutions $(v(x)=0)$, the minimum (Min), maximum (Max), average (Avg), and Stds of E_{tot} for five replications. For constraint handling, jEDE, CMA-ES, and GA reported feasible results in all the replications, whereas RBFOpt found two, and PSO reported one feasible solution. For the algorithms which reported infeasible solutions, Max, Avg, and Std values were also higher than the ones which reported feasible results in all replications because the penalised fitness function remained until the end of the optimisation. Regarding feasible alternatives, jEDE presented promising results with the lowest E_{tot} and Std values. The CMA-ES algorithm also suggested promising searchability for discovering feasible solutions, while finding slightly higher values of E_{tot} and Std than jEDE. Despite the feasible solutions, GA had higher E_{tot} when compared to jEDE and CMA-ES. During the optimisation process, the *CPU* of the algorithms was recorded as 82, 269, 819, 124, and 334 min for jEDE, CMA-ES, RBFOpt, GA, and PSO, respectively. This suggested that the jEDE was the most robust algorithm because its lower *CPU*, E_{tot}, and Std values than the other algorithms. The convergence graphs of all algorithms for this optimisation round are given in Fig. 5.17.

TABLE 5.8 Overview of the optimisation results for 5 replications at the building scale

Algorithm	$v(x)=0$	Min E_{tot}	Max E_{tot}	Avg E_{tot}	Std
jEDE (Pop size: 50)	5 out of 5	3072.8	3240.8	3163.5	61.1
jEDE (Pop size: 100)	5 out of 5	3182.5	3403.2	3296.1	71.0
CMA-ES	5 out of 5	3217.9	3404.0	3338.6	66.9
RBFOpt	2 out of 5	5165.1	74,456.9	41,559.0	29,261.3
GA (Pop size: 50)	5 out of 5	3934.1	4241.4	4134.6	109.3
GA (Pop size: 100)	5 out of 5	4011.8	4401.5	4253.1	129.7
PSO (Swarm size: 50)	1 out of 5	3384.6	42,463.4	29,392.7	13,939.7
PSO (Swarm size: 100)	1 out of 5	3330.3	41,652.1	23,312.7	14,979.3

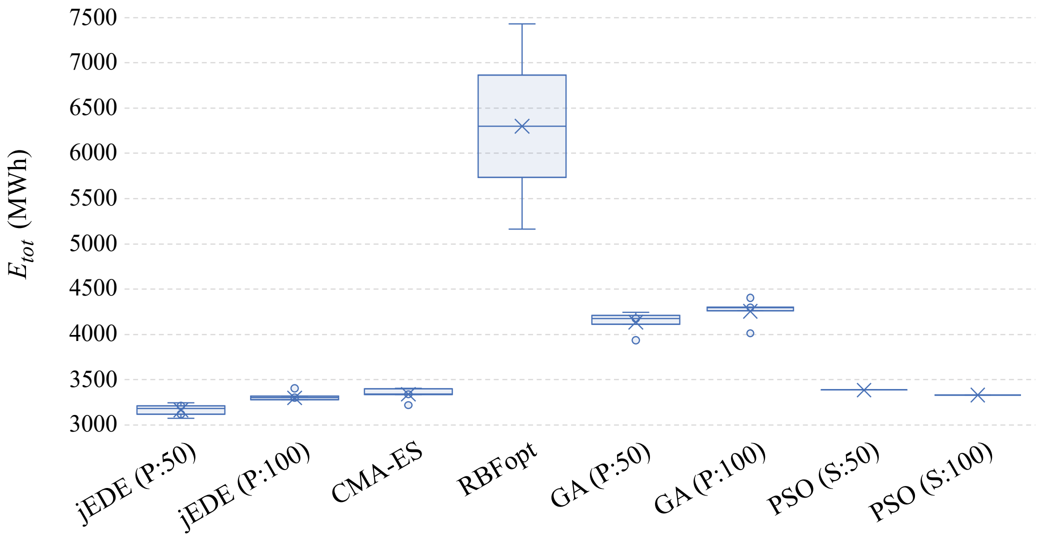


FIG. 5.16 Boxplots of the feasible solutions

The results of the jEDE with a population size of 50 was selected for further analysis of self-sufficiency at the building scale. When the design of the optimised complex was investigated in detail, weekly averages of air temperatures were observed as being between 19.5 °C and 25.1 °C in all zones. Since the violation was 0 at the end of the optimisation process, all the 36 DF values were higher than 2%, while 66 tons of lettuce was provided in one year. For energy performance, the E_{tot} of the optimised design was reported as 3072.8 MWh y^{-1}, while the values of E_R, E_F, and E_g were 4206.3 MWh y^{-1}, 1258.6 MWh y^{-1}, 2392.1 MWh y^{-1}, respectively.

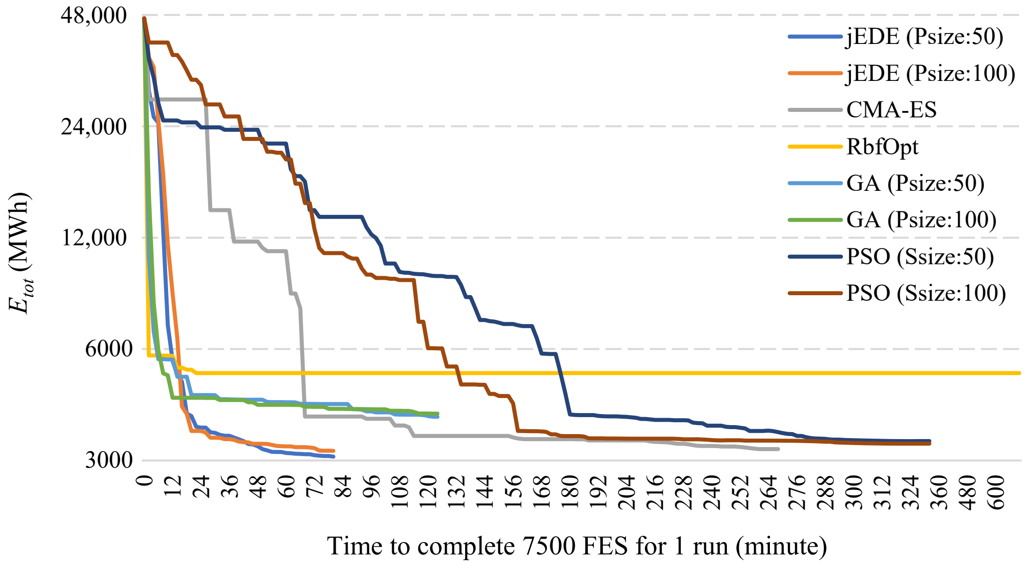


FIG. 5.17 Convergence of the algorithms in time to complete 7500 FES

Therefore, 43.7% self-sufficiency was achieved in energy, while 100% self-sufficiency was reached in lettuce production. Considering CoP as 1, E_R, E_F, and E_g were reported as 6854.9 MWh y^{-1}, 2246.4 MWh y^{-1}, 2392.1 MWh y^{-1}, respectively. Average air temperatures of three zones for each tower are given in Fig. 5.18.

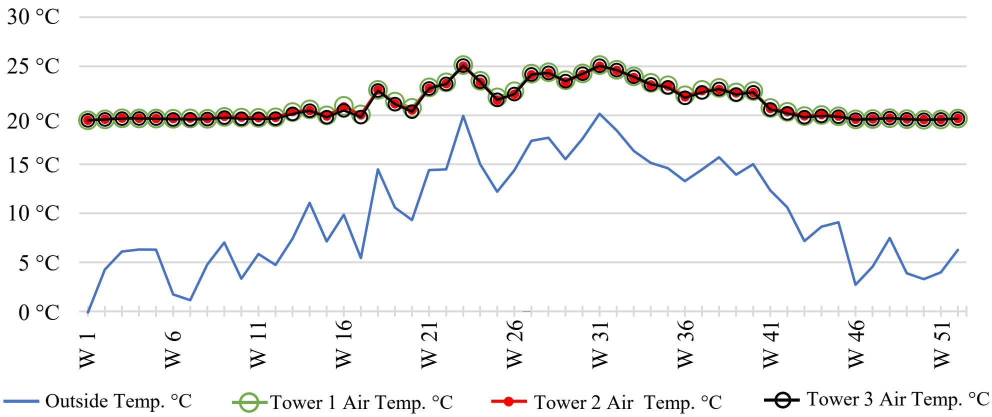


FIG. 5.18 Average air temperatures of the optimised Europoint complex

Finally, the efficiency of the MUZO methodology was tested by generating typical high-rise scenarios, which had the same set of design parameters for the complex instead of differentiating them as in the optimised solution. 7776 typical design scenarios were generated using the combinations of the parameters given in Table 5.9 for each tower. Design scenarios having more than two farming floors in the complex were discarded to have the same lettuce production for 1800 residents. Fig. 5.19 presents the comparison between the optimised design and typical scenarios, whereas Fig. 5.20 illustrates the parameter values of the optimised design and the best typical scenario.

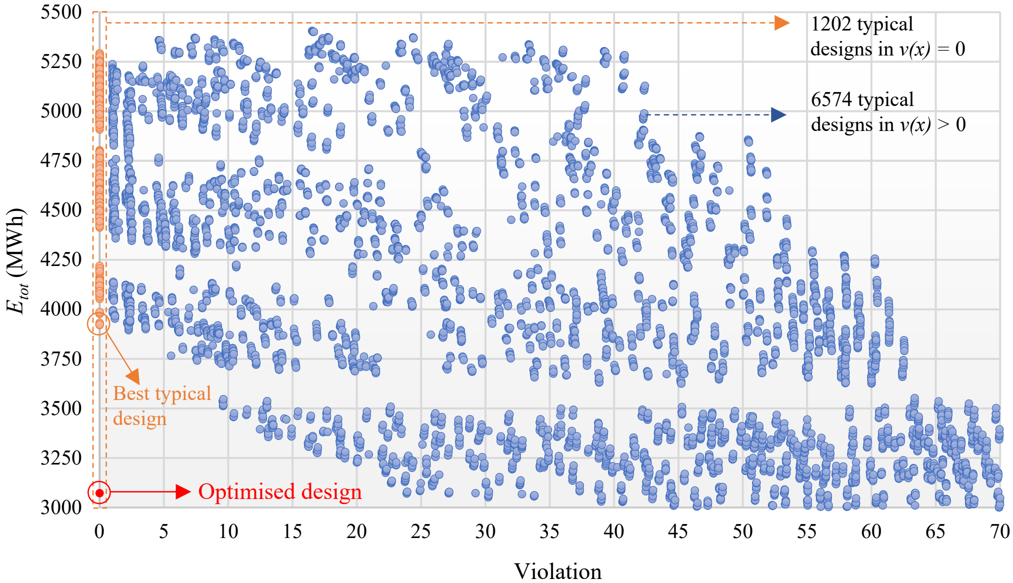


FIG. 5.19 MUZO design versus 7776 typical scenarios

TABLE 5.9 Parameter values used to generate typical scenarios

Tower #	Number of Farming Floors	Extrusion of Farming BIPV	Extrusion of Roof BIPV	Glazing Type	Shading Reflectance	Shading Distance	Window Reduction Size
1	[0, 1, 2]	[0, 12, 24]	[0, 3, 5]	[1, 4, 8, 12]	[0.3, 0.6, 0.9]	[0.25, 0.75, 1.50]	[0.0, 0.3, 0.6, 0.9]
2	"	[0, 5, 10]	"	"	"	"	"
3	"	[0, 10, 20]	"	"	"	"	"

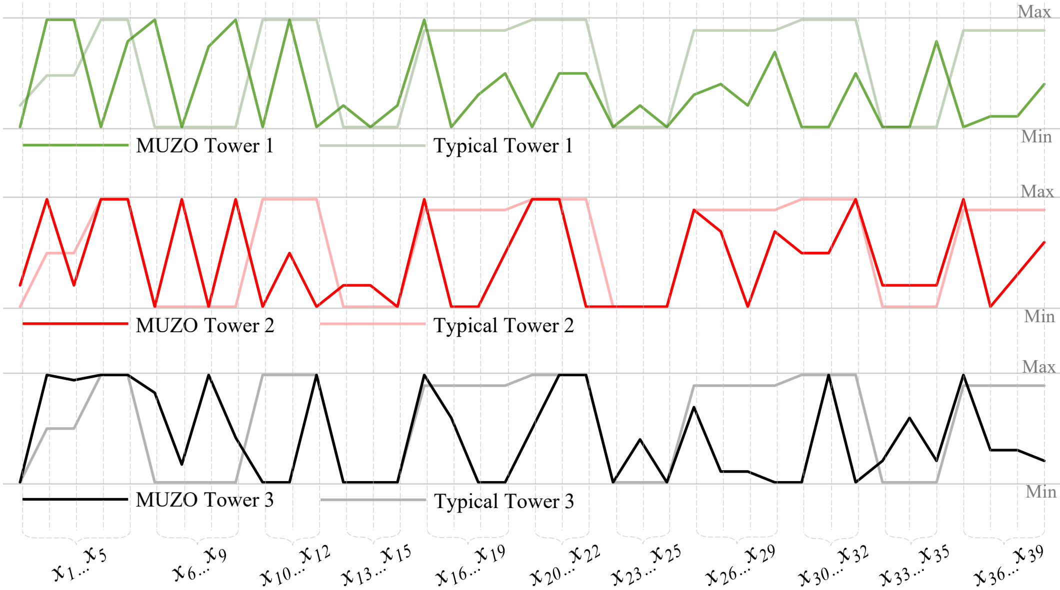


FIG. 5.20 Parameters of MUZO design and the best typical scenario

Results indicated that the studied problem was extremely challenging because of finding feasible solutions for DF in each orientation of nine zones while minimising E_{tot} . When the distribution of the typical scenarios was examined in Fig. 5.19, it was observed that there were 1202 feasible solutions out of 7776. Although some infeasible alternatives had similar E_{tot} values with the optimised solution, their DF values were less than 2%, which was lower than the Dutch building standards. Additionally, the lowest energy consumption in the typical scenarios was observed as 3922.9 MWh y^{-1}. When the optimised solution was compared to a typical design, the performance improvement was noted as 21%, corresponding to 850.1 MWh y^{-1} less energy consumption for the entire complex. The final design achieved at the end of the MUZO methodology is illustrated in Fig. 5.21 using the parameter values of the jEDE algorithm with a population size of 50.

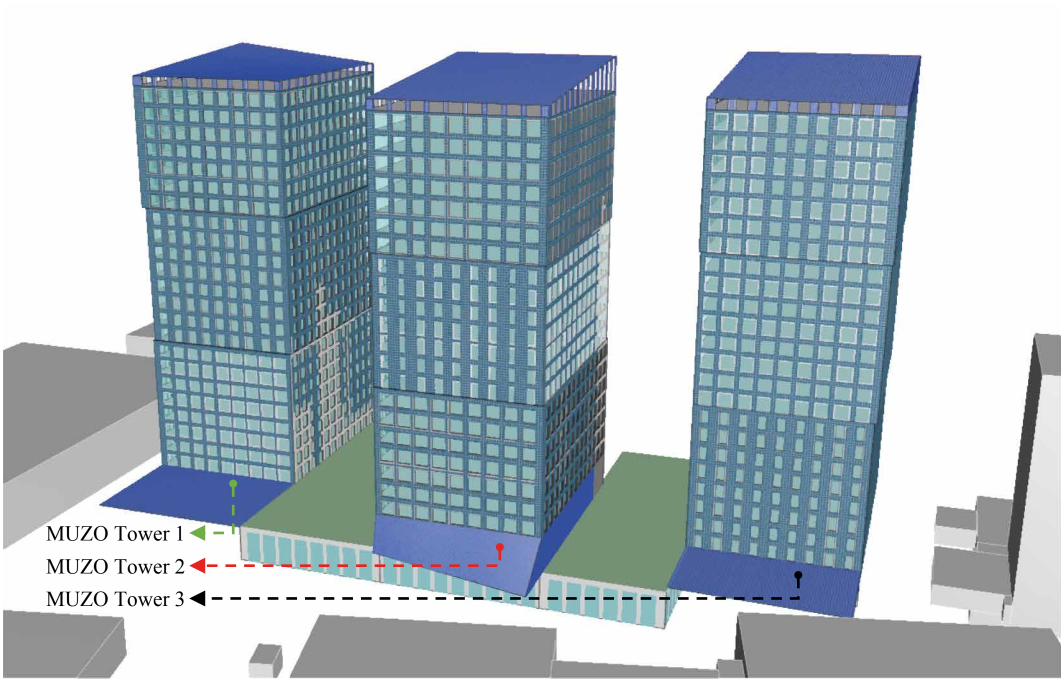


FIG. 5.21 Illustration of the optimised design for self-sufficiency at the building scale

5.3.3.2 Neighbourhood scale

Results for the neighbourhood scale were also conducted for 7500 FES using a computer with an Intel I7 5820 K core processor at 3.30 GHz, with 16-GB DDR4 of memory, and a 256-GB solid-state drive. The 7500 FES was defined as the maximum iteration count for the RBFMopt algorithm in the Opossum plug-in because of its non-populated application. In other algorithms, 50 population sizes were considered for jEDE, NSGA-II, HypE, SPEA-2, and MOEA/D with 150 generations, and 50 swarm sizes for NSPSO, and MACO with 150 iterations. The criteria for comparing the optimisation results of this problem were defined as finding feasible solutions for 36 constraints, having a large number of non-dominated solutions, and having a small *CPU*. Results of the jEDE algorithm were based on the stepwise run using incremental violation for F_p. Table 5.10 presents the overview of the optimisation results at the neighbourhood scale. Fig. 5.22 illustrates the search space and the lowest E_{tot} (non-dominated) results of jEDE, whereas Fig. 5.23 presents the same results for the rest of the other multi-objective algorithms.

TABLE 5.10 Overview of the optimisation results for 1 replication at the neighbourhood scale

Algorithm	Number of Non-Dominated Solutions in $(v(x)=0)$	Number of Non-Dominated Solutions Outperformed Other Algorithms	*CPU*
jEDE (stepwise)	30 out of 30	24 out of 30	1 d 21 h 14 m (for 30 F_p)
NSGA-II	20 out of 30	0 out of 30	1 h 58 m (for 1 run)
HypE	20 out of 30	6 out of 30	1 h 41 m (for 1 run)
SPEA-2 (Alt Pm)	30 out of 30	0 out of 30	2 h 36 m (for 1 run)
SPEA-2 (Pm)	28 out of 30	0 out of 30	2 h 36 m (for 1 run)
MACO	0 out of 30	0 out of 30	5 h 34 m (for 1 run)
NSPSO	5 out of 30	0 out of 30	5 h 35 m (for 1 run)
RBFMopt	0 out of 30	0 out of 30	2 d 10 h 10 m (for 1 run)
MOEA/D	0 out of 30	0 out of 30	5 h 32 m (for 1 run)

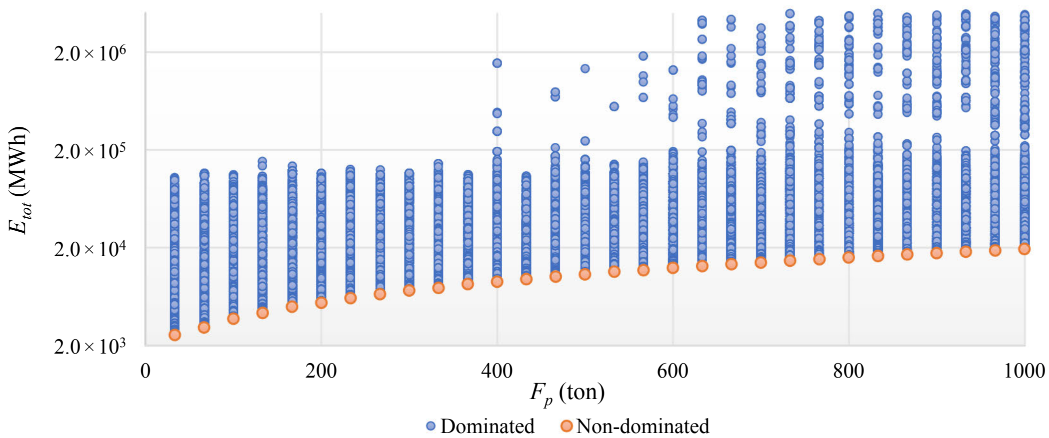


FIG. 5.22 Search space and optimisation results of jEDE for stepwise run

During the stepwise optimisation, jEDE discovered feasible solutions in all problems that corresponded to 30 non-dominated solutions. Because of the single-objective design of jEDE, the total run time was higher than other multi-objective optimisation algorithms owing to the 30 runs completed in each step. Contrarily, results of jEDE were used as a benchmark, because of its promising searchability. Results indicated that jEDE found 24 lower E_{tot} values than other algorithms. In other words, 24 results dominated the other non-dominated solutions discovered by multi-objective optimisation algorithms. On the other hand, HypE dominated six of the jEDE solutions, slightly.

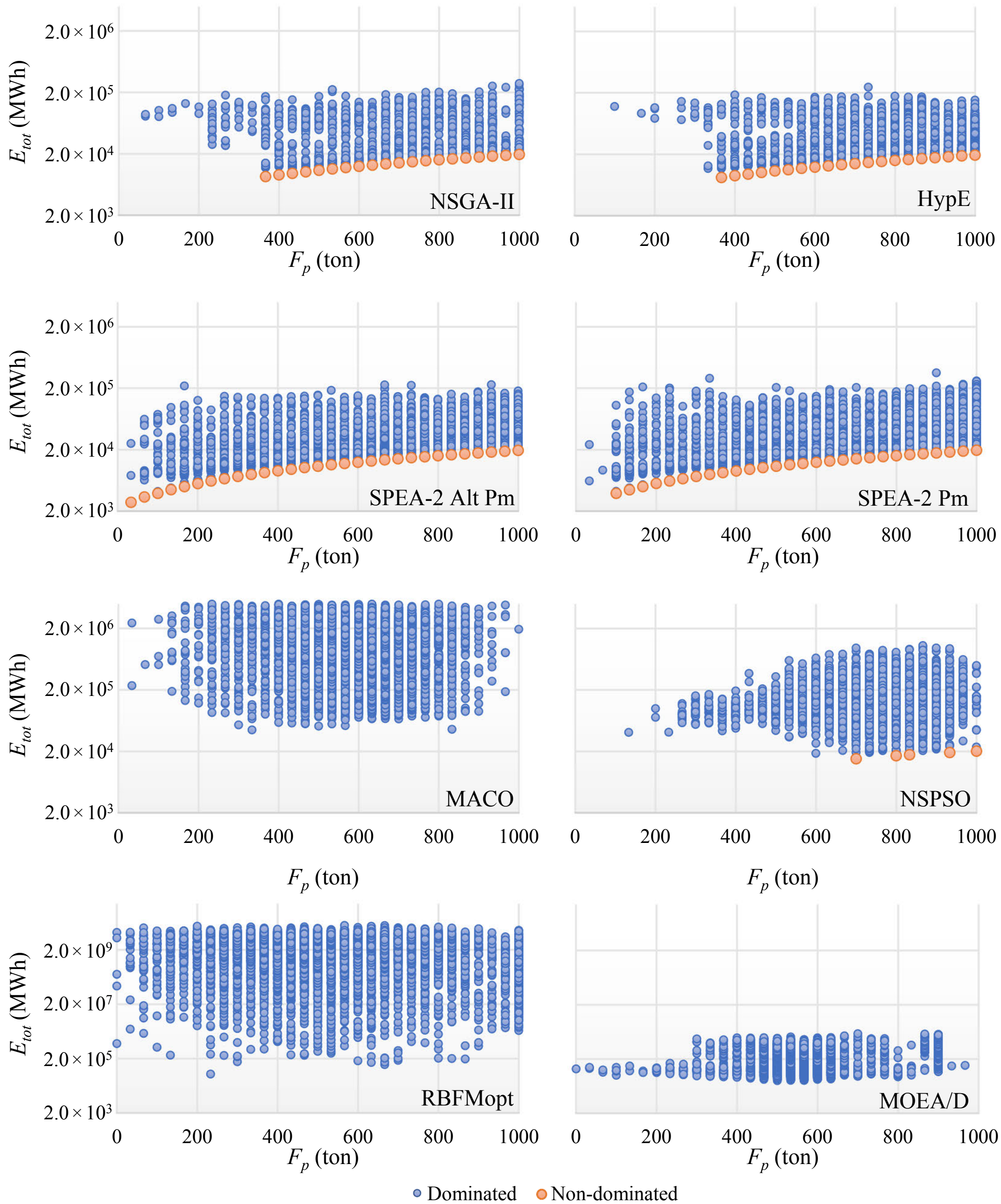


FIG. 5.23 Search space and optimisation results of multi-objective algorithms

Despite promising results in E_{tot}, NSGA-II and HypE could find 20 non-dominated solutions out of 30. However, the non-dominated solutions of NSGA-II were dominated by jEDE and HypE. SPEA-2 Alt. Pm. and Pm. applications discovered 30 and 28 non-dominated solutions, which were also dominated by jEDE and HypE, respectively. Additionally, NSPSO presented a limited number of feasible non-dominated solutions, whereas the MACO, RBFMopt, and MOEA/D algorithms discovered only infeasible alternatives. Therefore, jEDE, NSGA-II, HypE, and versions of the SPEA-2 algorithms were selected to investigate the potential of the Europoint complex in detail.

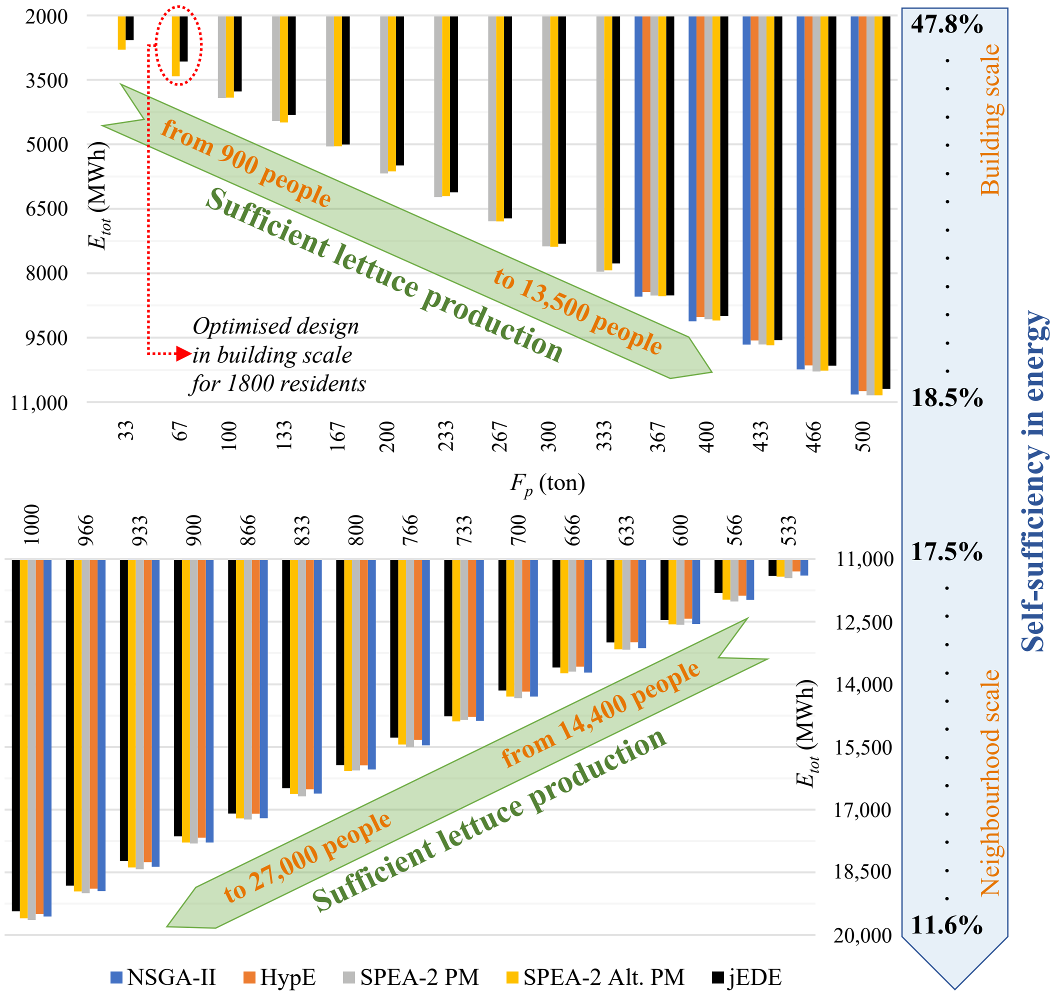


FIG. 5.24 Self-sufficiency potential of Europoint complex for neighbourhood scale

Fig. 5.24 illustrates the combined version of the non-dominated solutions, which have an E_{tot} between 2575.4 MWh y^{-1} and 19,639.6 MWh y^{-1} and an F_p between 33 tons and 1000 tons, as a result of the selected algorithms. The developed model indicated potential annual self-sufficiency in lettuce production starting from 900 people up to 27,000 people, whereas potential self-sufficiency in energy was observed for between 47.8% and 11.6% starting from the building scale to the neighbourhood scale.

Despite the increasing surface area of BIPV panels at higher values of F_p, self-sufficiency in energy was decreased because of the significant energy use of the closed farming systems. Regarding the energy results, variances of E_{tot} were higher at the building scale when compared to the neighbourhood scale. The reason was that the lower values of F_p increased the significance of the design parameters related to the energy consumption of the residential floors. In solutions with higher F_p values at the neighbourhood scale, the energy consumption of the closed farming systems dominated the impact of those parameters. Therefore, jEDE significantly improved upon other algorithms at the building scale, whereas jEDE and HypE slightly improved upon NSGA-II and SPEA-2 at the neighbourhood scale.

From a broad perspective of the search space, a linear correlation was observed between E_{tot} and F_p that was expected as the constant energy consumption and production results of the farming systems were associated with the number of the farming floors (x_1). To evaluate the potential impact of the developed model in Rotterdam city, the density of the habitation was considered at 3043 residents per km^2 [77].

A self-sufficiency map is illustrated in Fig. 5.25 using Stamen Maps [78].
For the highest value of F_p, the Europoint complex could provide lettuce for 27,000 residents in a 1.67 km radius that corresponded to 2.66% of the population of Rotterdam (assuming the city has 1,012,017 residents [77]).
To become a self-sufficient city after achievement of self-sufficiency at the neighbourhood scale, approximately 940,000 m^2 of vertical farming area would be needed to respond to the lettuce demand of the citizens of Rotterdam. In other words, 38 complexes like the Europoint complex, which involves 30 F_p levels that have 833 m^2 floor area each, would make Rotterdam self-sufficient in lettuce crop.

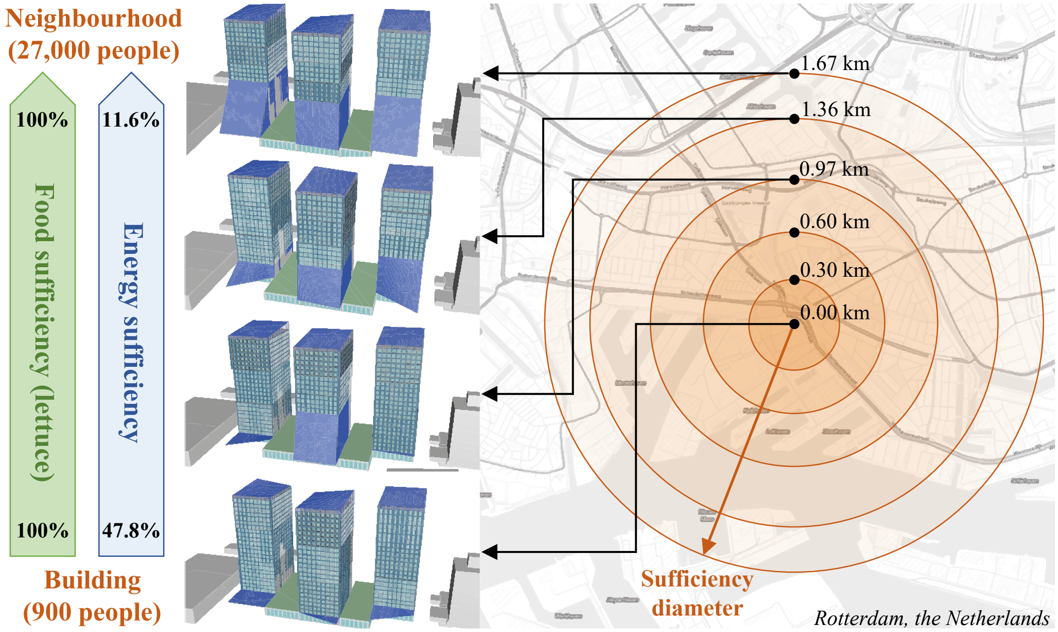


FIG. 5.25 Sufficiency diameter of Europoint complex in Rotterdam, the Netherlands

5.3.4 Discussion

This section presents the discussion based on the results of the sampling, ML, and optimisation experiments conducted for the Europoint complex. Five topics are discussed, which are defined as: the parametric high-rise model; ML for performance prediction; computational optimisation; self-sufficiency in energy and food; and the MUZO methodology for self-sufficient high-rise buildings in future cities.

1 Parametric high-rise model

Samples of Europoint complex were collected separately for each subdivision (zone) using a computer with an Intel I7 5820 K core processor at 3.30 GHz, with 16-GB DDR4 of memory, and a 256-GB solid-state drive. During the sampling process, the computer terminated the calculation process because of a memory shortage when the entire complex was simulated using 21 models. The same test was replicated using a computer with an Intel Xeon E5-2640 v4 core processor at 2.40 GHz, that had 64-GB DDR4 of memory, and a 1024-GB solid-state drive. The completion time was recorded as more than 1 h when it was expected to be approximately 22 m for the hourly simulation period. After the examination, the reason was defined as the data transfer between the simulation plug-ins and engines. Although less effort was required to conduct the results using all the simulation models of the complex, each sampling process was completed separately, which was three times more efficient than considering a sampling process for the entire complex. Lower simulation periods, i.e., 15 min, can cause an exponential increase in the efficiency of conducting the sampling results. In this paper, self-sufficiency in energy and food was examined subject to daylight performance. In the case of integrating other performance aspects related to self-sufficiency or comfort, the MUZO methodology would still be a feasible solution because of how it deals with different parts of the buildings as different design problems. Moreover, there was no error reported during the simulations because the studied building had an orthogonal floor plan and façade configuration. In the case of convex or nonconvex surfaces involved in any part of the building, errors or exponential increases in simulation time could be observed.

2 ML for performance prediction

45 surrogate models were developed to predict the performance aspects. While nine of these models were used to predict energy-related criteria, 36 of them were considered for daylight evaluation. Instead of using a high number of surrogate models to predict the daylight in detail, average values for each tower were considered during the initial phase of the ANN development. Despite the promising prediction accuracies, which had R^2 values higher than 0.8, it was observed that the minimum daylight requirement could not be achieved in all orientations of the three zones. Therefore, different surrogate models were considered that caused a slight increase in function evaluation during optimisation, but also a higher accuracy in terms of correct prediction. One may argue that a possible alternative could be to develop daylight models for average and deviatory values, which were not investigated owing to the limitations of the study. On the other hand, because of the full automation between the developed Python program and the predictive models developed in GH, extra effort was not needed to cope with a high number of surrogate models. Results of the grid search process indicated that different hyperparameter sets were required to predict performance with a high level of accuracy. This once again underlines the importance of grid search investigations for predicting performance aspects in the building simulation domain. When large numbers of ANN models are required for fitting during the grid search process, GPU usage can be considered not only for the studied problem scale in this paper but also for the problems focusing on larger scales in the built environment.

3 Computational optimisation

For the building scale, jEDE with the Optimus plug-in presented the most robust search behaviour because of it having the lowest fitness, *CPU*, and Std values. One reason could be that updating the number sliders in the GH environment requires additional time for each function evaluation. Another reason could be related to the procedures considered in optimisation algorithms, i.e., covariance matrix adaptation and the radial basis function. Despite updating the number sliders, GA in the Galapagos plug-in discovered near-optimal alternatives in less time when compared to other algorithms. On the other hand, CMA-ES presented promising solutions in terms of fitness and Std but required an expensive computation budget. Despite using the radial basis function, RBFopt could not perform desirable solutions that might be related to the high number of decision variables. This underlines once again an ongoing discussion for using either model-based algorithms (e.g., RBFopt) or optimisation procedures with predictive models (i.e., this paper) in architectural design [14]. The results of this paper indicated that surrogate-based optimisation algorithms are convenient to utilise in small-scale architectural design problems, whereas optimisation with surrogate models should be considered for design problems having an enormous number of design parameters. Therefore, an extensive investigation could be possible for large-scale design problems as attempted in this study. Since 7500 FES were sufficient for jEDE and CMA-ES, an investigation of a higher number of FES, which might slightly improve the results of other algorithms, was not considered. For the neighbourhood scale, the same number of FES was also considered for all algorithms. A higher number of FES with additional runs could also result in an increase in the number of conducted Pareto-front solutions. Since the purpose of the multi-objective formulation was to present the potentials of the developed model, this investigation remains limited. Additionally, the HypE and NSGA-II algorithms and variants of the SPEA-2 algorithm presented non-dominated results in an acceptable amount of time, whereas the MACO, NSPSO, RBFMopt, and MOEA/D algorithms could not provide promising results and required an expensive computational budget. Moreover, stepwise jEDE results indicated that results of the multi-objective optimisation algorithms could be dominated in 24 solutions out of 30. This highlights a gap in the development of multi-objective optimisation algorithms, which are capable of coping with high-dimensional constrained design problems in the architecture domain. Using predictive models, 577,500 FES are completed during the optimisation processes in 3 weeks. This task would take more than 19 years if simulation-based optimisation was considered for the same number of FES. This reminds us once again of the importance of selecting convenient optimisation methods during the conceptual phase of the design process.

4 Self-sufficiency in energy and food

Considering the assumption concerning lettuce consumption, that 1800 people live in the three towers of Europoint complex, the E_{tot} of the optimised design at the building scale was reported as 3072.8 MWh y^{-1} (for CoP 5) while E_R, E_F, and E_g were 4206.3 MWh y^{-1}, 1258.6 MWh y^{-1}, 2392.1 MWh y^{-1}, respectively. Despite the detailed parameter investigations of glazing types, window sizes, the shapes of the BIPV surfaces, and the shading devices used for each orientation in each tower, 43.7% sufficiency could be reached in terms of electricity usage at the building scale. Moreover, the required energy consumption for self-sufficiency at the neighbourhood scale reached up to 19,639.6 MWh y^{-1} because of the providing of lettuce crops for 27,000 people using 30 floors for vertical farming. Considering that one floor of farming requires 629.3 MWh y^{-1}, 96.1% of the energy demand was from food production for providing 1000 tons of lettuce for the 27,000 people. Although the energy demand of vertical farming was decreased when fewer farming floors were considered in the entire complex, more than 10 times the energy use intensity was required when compared to the residential floors. Even though the benefits of closed farming systems for food production in the centre of the city with low CO_2 emissions, high energy consumption remains one of the big challenges. This highlights the necessity of integral designs that use the potential solar and wind power of the building environment, as well as the importance of combining vertical farming with urban farming on roof-tops and in unused parts of the city to achieve self-sufficiency with lower energy consumption at the neighbourhood scale.

5 MUZO methodology for self-sufficient high-rise buildings in future cities

The MUZO methodology was utilised to conduct the self-sufficient design alternatives for energy consumption and food production (demonstrating sufficiency for lettuce crops), focusing on the Europoint complex in Rotterdam. Various advantages of considering this methodology were observed (e.g., coping with complex simulation models for sampling, investigation ability of the best prediction accuracy for multiple performance aspects, and the extensive investigation of the search space that leads to comprehensive decision-making in the design process because of employing multiple optimisation algorithms with replications). When compared to typical scenarios using the same parameter sets for the entire high-rise design, the energy consumption discovered by the MUZO methodology was improved by 21%. For self-sufficiency in food production, results suggested various floor selections on which to place the closed farming system, considering the total energy consumption as well as the total energy generation. Because of efficient agricultural production in closed farming systems, residents of the Europoint complex and habitants in the neighbourhood (up to 27,000) people could exploit the lettuce production in the Europoint complex. The self-sufficiency of lettuce crops for food production was demonstrated in this study. The variety in agricultural products could be increased by considering different simulation parameters to provide necessary indoor conditions for different crops using the same computational model. However, self-sufficiency in energy was not as achievable as food production. The results indicated that 100% self-sufficiency in energy could not be achieved at the building scale using existing BIPV technology. Therefore, combined systems for BIPV, battery, wind [27], and combined heat and power systems can be considered to improve the self-sufficiency at the building scale. Moreover, considering the potential of other buildings in the surrounding area (e.g., using their roofs for PV panels) appears to be a vital approach for future cities. The involvement of the surrounding buildings in addition to the buildings under study increases the complexity of the design problem. Considering the MUZO methodology can be a feasible approach, as it was able to present promising results for the three high-rise buildings in this paper.

5.4 Conclusion

This paper presents the results of an optimisation investigation of high-rise buildings for self-sufficiency in terms of energy consumption and food production for lettuce crops using AI techniques. The utilisation of MUZO for the Europoint complex located in Rotterdam managed the sampling process for 9000 design alternatives and the development of 45 ANN models, which considered grid searches for five hyperparameters in the first two phases of the methodology. The final phase focussed on optimisation of the three towers by employing the developed surrogate models with 117 decision variables, using 13 optimisation algorithms with single-objective and multi-objective problem formulations, which were subject to 37 and 36 constraints for the building and neighbourhood scales. The results showed that 100% self-sufficiency in food production for lettuce crops, and 43.7% self-sufficiency in energy consumption can be reached at the building scale, including 1800 residents. At the neighbourhood scale, sufficient lettuce production can be provided for up to 27,000 people living in a radius of 1.67 km by decreasing self-sufficiency in energy up to 11.6%. The relevance of the MUZO methodology is also shown by its discovery of self-sufficient high-rise alternatives at the building scale with an improvement of 21% in the considered performance aspects when compared to typical design scenarios. Self-sufficiency scores achieved at the building and neighbourhood scales highlight the necessity of integrating the potentials of the surrounding buildings in addition to the high-rise buildings in question. Hence, self-sufficient cities can be achieved by developing self-sufficient neighbourhoods.

- Limitations of the study: The importance of this study for future cities is presented in the digital (virtual) environment by the obtained ML scores, optimisation results, and the validation of the utilised MUZO methodology. Validation between the results obtained from the digital environment and the monitored data from the indoor environment of the real applications is one of the limitations of this study. Regarding the HVAC system, the default setup of the HB and LB plugins is considered during the energy simulations. A detailed HVAC setup (e.g., condenser or VRF loops) may improve the optimisation results using other plug-ins of LB tools. Additionally, only lettuce crops are integrated into the closed farming system, as they are one of the most well-documented agricultural products for vertical farming. Nevertheless, the developed computational model has the potential to consider other agricultural products by changing the parameters of the energy model to provide the necessary indoor conditions for the growing process of other plants. Although plant factories use water resources effectively when compared to traditional farming, self-sufficiency in water may require additional design and operational parameters to be considered in the optimisation process. Finally, the urban heat island impact of the proposals, which may be the primary focus of future works, is not taken into consideration.

- Future works: Higher self-sufficiency scores at the neighbourhood scale may require the use of random forests [79] and convolutional neural networks [80] for detecting the potential of roof spaces for PV panels. Considering self-sufficiency in different time frames (i.e., monthly or weekly) can provide an overview of the sufficiency performance in different weather conditions. Since the climate region has an inevitable impact on self-sufficiency aspects, hypothetical models developed for various climate zones may guide the necessary actions to be taken while transforming our cities for a sustainable future. In this respect, providing a potential reduction in CO_2 emissions, in addition to energy consumption and food production, may help policymakers to develop self-sufficient, carbon-neutral, energy-positive lives in metropolises. Energy models can also consider different set points as variables and various occupancy scenarios to decrease consumption. The impact of creating separate farming buildings instead of considering farming and residential levels in a single building may also provide a better operation in different scenarios. Integrating daylight simulation into the energy model may also decrease the total energy demand, with an additional computation budget owing to the hourly illuminance data. In addition to daylight, optimising the view of the residential spaces can be an added value that may increase the demand for self-sufficient high-rise buildings owing to the promising city view. Using the same model, the diversity of the agricultural products can be increased to include leafy greens, vine crops, and tomatoes [7], in addition to integrating other self-sufficiency aspects (e.g., harvesting water or adding ducted openings for wind energy [81]). Involving parameters of the façade design of the closed farming systems [23], and semi-transparent PV panels can reduce the energy demand while increasing the complexity of the optimisation problem. At the neighbourhood scale, different types of food production systems can find a place in various locations that allow for the provision of a large amount of various crops. Hence, the potential of the neighbourhood scale can be involved in providing vital food products to achieve self-sufficient cities in the future. Finally, economic aspects considering the fundamentals of the circular built environment may present an additional long-term strategy for decreasing the life-cycle cost and CO_2 emissions of future cities.

Data availability

Datasets related to this article can be found at (an open-source online data repository hosted at 4TU Research Data [76]): https://doi.org/10.4121/17129420.v1

References

[1] Ali, M.M.; Al-Kodmany, K. Tall Buildings and Urban Habitat of the 21st Century: A Global Perspective. *Buildings* **2012**, *2*, 384–423, https://doi.org/10.3390/buildings2040384.

[2] Godoy-Shimizu, D.; Steadman, P.; Hamilton, I.; Donn, M.; Evans, S.; Moreno, G.; Shayesteh, H. Energy use and height in office buildings. *Build. Res. Inf.* **2018**, *46*, 845–863, https://doi.org/10.1080/09613218.2018.1479927.

[3] Benke, K.; Tomkins, B. Future food-production systems: Vertical farming and controlled-environment agriculture. *Sustain. Sci. Pract. Policy* **2017**, *13*, 13–26, https://doi.org/10.1080/15487733.2017.1394054.

[4] Food and Agriculture Organization of the United Nations (FAO). Database on Arable Land 2016. Available online: http://data.worldbank.org/indicator/AG.LND.ARBL.HA.PC?end%20&hx003D;2013&hx0026;start%20&hx003D;1961&hx0026;view&hx003D;chart (accessed on 10 November 2021).

[5] Graamans, L.; Baeza, E.; van den Dobbelsteen, A.; Tsafaras, I.; Stanghellini, C. Plant factories versus greenhouses: Comparison of resource use efficiency. *Agric. Syst.* **2018**, *160*, 31–43, https://doi.org/10.1016/j.agsy.2017.11.003.

[6] Kozai, T.; Ohyama, K.; Chun, C. Commercialized closed systems with artificial lighting for plant production. In Proceedings of the V International Symposium on Artificial Lighting in Horticulture 711, 30 June **2006**, Lillehammer, Norway.

[7] Zeidler, C.; Schubert, D.; Vrakking, V. Vertical Farm 2.0: Designing an Economically Feasible Vertical Farm-A combined Eu-ropean Endeavor for Sustainable Urban Agriculture. Assoc. Vert. Farming **2017**. Available online: https://elib.dlr.de/116034/ (accessed on 4 October 2021)

[8] Imam, M.; Kolarevic, B. Towards Resource-Generative Skyscrapers. *Int. J. High.-Rise Build.* **2018**, *7*, 161–170.

[9] Voss, K.; Musall, E.; Lichtmeß, M. From Low-Energy to Net Zero-Energy Buildings: Status and Perspectives. *J. Green Build.* **2011**, *6*, 46–57, https://doi.org/10.3992/jgb.6.1.46.

[10] Vale, B.; Vale, R. *The New Autonomous House: Design and Planning for Sustainability*; Thames & Hudson: London, UK, **2000**.

[11] Evins, R. A review of computational optimisation methods applied to sustainable building design. *Renew. Sustain. Energy Rev.* **2013**, *22*, 230–245, https://doi.org/10.1016/j.rser.2013.02.004.

[12] Samuelson, H.; Claussnitzer, S.; Goyal, A.; Chen, Y.; Romo-Castillo, A. Parametric energy simulation in early design: High-rise residential buildings in urban contexts. *Build. Environ.* **2016**, *101*, 19–31, https://doi.org/10.1016/j.buildenv.2016.02.018.

[13] Ekici, B.; Kazanasmaz, Z.T.; Turrin, M.; Taşgetiren, M.F.; Sariyildiz, I.S. Multi-zone optimisation of high-rise buildings using artificial intelligence for sustainable metropolises. Part 1: Background, methodology, setup, and machine learning results. *Sol. Energy* **2021**, *224*, 373–389, https://doi.org/10.1016/j.solener.2021.05.083.

[14] Ekici, B.; Kazanasmaz, Z.T.; Turrin, M.; Taşgetiren, M.F.; Sariyildiz, I.S. Multi-zone optimisation of high-rise buildings using artificial intelligence for sustainable metropolises. Part 2: Optimisation problems, algorithms, results, and method validation. *Sol. Energy* **2021**, *224*, 309–326, https://doi.org/10.1016/j.solener.2021.05.082.

[15] Gan, V.J.; Wong, H.; Tse, K.T.; Cheng, J.C.; Lo, I.M.; Chan, C.M. Simulation-based evolutionary optimization for ener-gy-efficient layout plan design of high-rise residential buildings. *J. Clean. Prod.* **2019**, *231*, 1375–1388.

[16] Li, Y.; Li, X. Natural ventilation potential of high-rise residential buildings in northern China using coupling thermal and airflow simulations. *Build. Simul.* **2014**, *8*, 51–64, https://doi.org/10.1007/s12273-014-0188-1.

[17] Raji, B.; Tenpierik, M.J.; Dobbelsteen, A.V.D. An assessment of energy-saving solutions for the envelope design of high-rise buildings in temperate climates: A case study in the Netherlands. *Energy Build.* **2016**, *124*, 210–221, https://doi.org/10.1016/j.enbuild.2015.10.049.

[18] Jayaweera, N.; Rajapaksha, U.; Manthilake, I. A parametric approach to optimize solar access for energy efficiency in high-rise residential buildings in dense urban tropics. *Sol. Energy* **2021**, *220*, 187–203, https://doi.org/10.1016/j.solener.2021.02.054.

[19] Wang, M.; Hou, J.; Hu, Z.; He, W.; Yu, H. Optimisation of the double skin facade in hot and humid climates through altering the design parameter combinations. In *Building Simulation*; Springer: Berlin/Helideberg, Germany, **2021**; pp. 511–521

[20] Chen, X.; Huang, J.; Zhang, W.; Yang, H. Exploring the optimization potential of thermal and power performance for a low-energy high-rise building. *Energy Procedia* **2019**, *158*, 2469–2474, https://doi.org/10.1016/j.egypro.2019.01.372.

[21] Chen, X.; Yang, H. A multi-stage optimization of passively designed high-rise residential buildings in multiple building oper-ation scenarios. *Appl. Energy* **2017**, *206*, 541–557.

[22] Chen, X.; Yang, H. Integrated energy performance optimization of a passively designed high-rise residential building in dif-ferent climatic zones of China. *Appl. Energy* **2018**, *215*, 145–158.

[23] Graamans, L.; Tenpierik, M.; van den Dobbelsteen, A.; Stanghellini, C. Plant factories: Reducing energy demand at high in-ternal heat loads through facade design. *Appl. Energy* **2020**, *262*, 114544.

[24] Chen, X.; Yang, H.; Peng, J. Energy optimization of high-rise commercial buildings integrated with photovoltaic facades in urban context. *Energy* **2019**, *172*, 1–17, https://doi.org/10.1016/j.energy.2019.01.112.

[25] Chen, X.; Huang, J.; Yang, H.; Peng, J. Approaching low-energy high-rise building by integrating passive architectural design with photovoltaic application. *J. Clean. Prod.* **2019**, *220*, 313–330, https://doi.org/10.1016/j.jclepro.2019.02.137.

[26] Giouri, E.D.; Tenpierik, M.; Turrin, M. Zero energy potential of a high-rise office building in a Mediterranean climate: Using multi-objective optimization to understand the impact of design decisions towards zero-energy high-rise buildings. *Energy Build.* **2019**, *209*, 109666, https://doi.org/10.1016/j.enbuild.2019.109666.

[27] Liu, J.; Wang, M.; Peng, J.; Chen, X.; Cao, S.; Yang, H. Techno-economic design optimization of hybrid renewable energy applications for high-rise residential buildings. *Energy Convers. Manag.* **2020**, *213*, 112868, https://doi.org/10.1016/j.enconman.2020.112868.

[28] Wolpert, D.H.; Macready, W.G. No free lunch theorems for optimization. *IEEE Trans. Evol. Comput.* **1997**, *1*, 67–82, https://doi.org/10.1109/4235.585893.

[29] Wood, A. Sustainability: A new high-rise vernacular? *Struct. Des. Tall Spec. Build.* **2007**, *16*, 401–410.

[30] Coit, D.W.; Smith, A.E. Penalty guided genetic search for reliability design optimization. *Comput. Ind. Eng.* **1996**, *30*, 895–904, https://doi.org/10.1016/0360-8352(96)00040-x.

[31] Smith, A.E.; Coit, D.W.; Baeck, T.; Fogel, D.; Michalewicz, Z. Penalty functions. *Handb. Evol. Comput.* **1997**, *97*, C5.

[32] Ekici, B.; Cubukcuoglu, C.; Turrin, M.; Sariyildiz, I.S. Performative computational architecture using swarm and evolutionary optimisation: A review. *Build. Environ.* **2019**, *147*, 356–371, https://doi.org/10.1016/j.buildenv.2018.10.023.

[33] Cubukcuoglu, C.; Ekici, B.; Tasgetiren, M.F.; Sariyildiz, S. OPTIMUS: Self-Adaptive Differential Evolution with Ensemble of Mutation Strategies for Grasshopper Algorithmic Modeling. *Algorithms* **2019**, *12*, 141, https://doi.org/10.3390/a12070141.

[34] Cichocka, J.M.; Migalska, A.; Browne, W.N.; Rodriguez, E. SILVEREYE–the implementation of Particle Swarm Optimization algorithm in a design optimization tool. In Proceedings of the International Conference on Computer-Aided Architectural Design Futures, Istanbul Technical University, 12–14 July **2017**; Springer: Singapore; pp. 151–169. https://doi.org/10.1007/978-981-10-5197-5_9.

[35] Rutten, D. Galapagos: On the Logic and Limitations of Generic Solvers. *Arch. Des.* **2013**, *83*, 132–135, https://doi.org/10.1002/ad.1568.

[36] Wortmann, T. Opossum-introducing and evaluating a model-based optimization tool for grasshopper. Proceedings of the 22nd CAADRIA Conference, 5-8 April 2017, Xi'an Jiaotong-Liverpool University, Suzhou, China, **2017**, pp. 283-292.

[37] Kennedy, J.; Eberhart, R. Particle swarm optimization. In Proceedings of the ICNN'95-International Conference on Neural Networks, Perth, WA, Australia, 27 November – 1 December 1995; IEEE: Piscataway, NJ, USA, **1995**; pp. 1942–1948.

[38] Goldberg, D.E. *Genetic Algorithms*; Pearson Education India: Delhi, NCR, Noida, **2006**.

[39] Hansen, N. The CMA evolution strategy: A comparing review. *Towards New Evol. Comput.* **2006**, *192*, 75–102.
[40] Costa, A.; Nannicini, G. RBFOpt: An open-source library for black-box optimization with costly function evaluations. *Math. Program. Comput.* **2018**, *10*, 597–629, https://doi.org/10.1007/s12532-018-0144-7.
[41] Makki, M.; Showkatbakhsh, M.; Tabony, A.; Weinstock, M. Evolutionary algorithms for generating urban morphology: Vari-ations and multiple objectives. *Int. J. Archit. Comput.* **2019**, *17*, 5–35.
[42] Vierlinger, R.; Hofmann, A. A Framework for Flexible Search and Optimization in Parametric Design. In Proceedings of the Rethinking Prototyping-Proceedings of the Design Modelling Symposium, University of the Arts, 28 September – 2 October, Berlin, Germany, **2013**.
[43] Deb, K.; Pratap, A.; Agarwal, S.; Meyarivan, T. A fast and elitist multiobjective genetic algorithm: NSGA-II. *IEEE Trans. Evol. Comput.* **2002**, *6*, 182–197.
[44] Bader, J.; Zitzler, E. HypE: An algorithm for fast hypervolume-based many-objective optimization. *Evol. Comput.* **2011**, *19*, 45–76.
[45] Zitzler, E.; Laumanns, M.; Thiele, L. *SPEA2: Improving the Strength Pareto Evolutionary Algorithm*; TIK-Report: Eidgenössische Technische Hochschule Zürich (ETH), Institut für Technische Informatik und Kommunikationsnetze (TIK), Zürich, Switzerland, **2001**; Volume 103, https://doi.org/10.3929/ethz-a-004284029.
[46] Wortmann, T.; Natanian, J. Multi-Objective Optimization for Zero-Energy Urban Design in China: A Benchmark. P*roc. Sim AUD* **2020**, 203–210.
[47] Li, X. A Non-dominated Sorting Particle Swarm Optimizer for Multiobjective Optimization. In Proceedings of the Genetic and Evolutionary Computation Conference, Chicago, IL, USA, July 12–16 2003; Springer: Berlin/Heidelberg, Germany; **2003**, pp. 37–48, https://doi.org/10.1007/3-540-45105-6_4.
[48] Gao, Y.; Guan, H.; Qi, Z.; Hou, Y.; Liu, L. A multi-objective ant colony system algorithm for virtual machine placement in cloud computing. *J. Comput. Syst. Sci.* **2013**, *79*, 1230–1242, https://doi.org/10.1016/j.jcss.2013.02.004.
[49] Zhang, Q.; Li, H. MOEA/D: A Multiobjective Evolutionary Algorithm Based on Decomposition. *IEEE Trans. Evol. Comput.* **2007**, *11*, 712–731, https://doi.org/10.1109/tevc.2007.892759.
[50] MOR, Solar Decathlon Team of TU Delft. Available online: https://mor.tudelft.nl/ (accessed on 18 December 2021).
[51] Solar Decathlon Europe. Available online: https://solardecathlon.eu/ (accessed on 18 December 2021).
[52] The European Parliament and the Council of the European Union. Amending Directive 2010/31/EU on the energy performance of buildings. *Off. J. Eur. Union L* **2018**, L 153 pp. 13-35.
[53] McNeel, R. Rhinoceros. NURBS Modleing for Windows: Available online: http://www.rhino3d.com/ (accessed on 15 June 2021).
[54] Rutten, D.; McNeel, R. *Grasshopper3D*; Robert McNeel & Associates: Seattle, WA, USA, **2007**.
[55] Roudsari, M.S.; Pak, M.; Smith, A. Ladybug: A parametric environmental plugin for grasshopper to help designers create an envi-ronmentally-conscious design. In Proceedings of the 13th International IBPSA Conference Held in Lyon, France, 26–28 August **2013**; pp. 3128–3135.
[56] Ward, G.J. The RADIANCE lighting simulation and rendering system. In Proceedings of the 21st Annual Conference on Computer Graphics and Interactive Techniques in Orlando, FL, USA, 24–29 July 1994; Association for Computing Machinery: New York, NY, USA; **1994**; pp. 459–472.
[57] Hammad, A.W.; Nezhad, A.A.; Grzybowska, H.; Wu, P.; Wang, X. Mathematical optimisation of location and design of windows by considering energy performance, lighting and privacy of buildings. *Smart Sustain. Built Environ.* **2019**, *8*, 117–137, https://doi.org/10.1108/sasbe-11-2017-0070.
[58] Wang, S.; Yi, Y.K.; Liu, N. Multi-objective optimization (MOO) for high-rise residential buildings' layout centered on daylight, visual, and outdoor thermal metrics in China. *Build. Environ.* **2021**, *205*, 108263, https://doi.org/10.1016/j.buildenv.2021.108263.
[59] Kirimtat, A.; Krejcar, O.; Ekici, B.; Tasgetiren, M.F. Multi-objective energy and daylight optimization of amorphous shading devices in buildings. *Sol. Energy* **2019**, *185*, 100–111, https://doi.org/10.1016/j.solener.2019.04.048.
[60] Ekici, B.; Kazanasmaz, T.; Turrin, M.; Tasgetiren, M.F.; Sariyildiz, I.S. A Methodology for daylight optimisation of high-rise buildings in the dense urban district using overhang length and glazing type variables with surrogate modelling. *J. Physics Conf. Ser.* **2019**, *1343*, 012133, https://doi.org/10.1088/1742-6596/1343/1/012133.

[61] Sepúlveda, A.; De Luca, F.; Thalfeldt, M.; Kurnitski, J. Analyzing the fulfillment of daylight and overheating requirements in residential and office buildings in Estonia. *Build. Environ.* **2020**, *180*, 107036, https://doi.org/10.1016/j.buildenv.2020.107036.
[62] Yi, Y.K. Building facade multi-objective optimization for daylight and aesthetical perception. *Build. Environ.* **2019**, *156*, 178–190, https://doi.org/10.1016/j.buildenv.2019.04.002.
[63] Lee, J.; Boubekri, M.; Liang, F. Impact of Building Design Parameters on Daylighting Metrics Using an Analysis, Prediction, and Optimization Approach Based on Statistical Learning Technique. *Sustainability* **2019**, *11*, 1474, https://doi.org/10.3390/su11051474.
[64] Epwmap—Ladybug Tools. Available online: https://www.ladybug.tools/epwmap/ (accessed on 10 June 2021).
[65] Loh, W.-L. On Latin hypercube sampling. *Ann. Stat.* **1996**, *24*, 2058–2080.
[66] Roman, N.D.; Bre, F.; Fachinotti, V.D.; Lamberts, R. Application and characterization of metamodels based on artificial neural networks for building performance simulation: A systematic review. *Energy Build.* **2020**, *217*, 109972, https://doi.org/10.1016/j.enbuild.2020.109972.
[67] Chatzikonstantinou, I.; Sariyildiz, S. Approximation of simulation-derived visual comfort indicators in office spaces: a com-parative study in machine learning. *Archit. Sci. Rev.* **2016**, *59*, 307–322.
[68] Bottou, L. Large-Scale Machine Learning with Stochastic Gradient Descent. In Proceedings of the 19th International Conference on Computational Statistics, Paris, France, 22–27 August **2010**; Springer: Berlin/Heildeberge, Germany; pp. 177–186, https://doi.org/10.1007/978-3-7908-2604-3_16.
[69] Srivastava, N.; Hinton, G.; Krizhevsky, A.; Sutskever, I.; Salakhutdinov, R. Dropout: A simple way to prevent neural networks from overfitting. *J. Mach. Learn. Res.* **2014**, *15*, 1929–1958.
[70] Van Rossum, G.; Drake, F. *Python 3 Reference Manual*; CreateSpace: Scotts Valley, CA, USA, **2009**.
[71] McKinney, W. Data structures for statistical computing in python. In Proceedings of the 9th Python in Science Conference, Austin, TX, USA, 28 June–3 July **2010**; pp. 51–56.
[72] Chollet, F. Keras. Available online: https://keras.io/ (accessed on 5 October 2021).
[73] Abadi, M.; Barham, P.; Chen, J.; Chen, Z.; Davis, A.; Dean, J.; Devin, M.; Ghemawat, S.; Irving, G.; Isard, M. Tensorflow: A system for large-scale machine learning. In Proceedings of the 12th {USENIX} Symposium on Operating Systems Design and Implementation ({OSDI} 16), 2–4 November, Savannah, GA, USA, USENIX Association, Berkeley, CA, USA, **2016**; pp. 265–283.
[74] Pedregosa, F.; Varoquaux, G.; Gramfort, A.; Michel, V.; Thirion, B.; Grisel, O.; Blondel, M.; Prettenhofer, P.; Weiss, R.; Dubourg, V. Scikit-learn: Machine learning in Python. *J. Mach. Learn. Res.* **2011**, *12*, 2825–2830.
[75] Taccari, L. Joyplots. Available online: https://github.com/sbebo/joypy/blob/master/Joyplot.ipynb Github repository (accessed on 10 September 2021).
[76] Ekici, B.; Turkcan, O.F.S.F.; Turrin, M.; Sariyildiz, I.S.; Tasgetiren, M.F. Multi-zone simulation results of Europoint complex for self-sufficiency in energy consumption and food production in Rotterdam. 4TU.ResearchData. Dataset, https://doi.org/10.4121/17129420.v1, **2021** (accessed on 19 December 2021)
[77] World Population Review. Available online: https://worldpopulationreview.com/world-cities/rotterdam-population (accessed on 12 November 2021).
[78] Stamen Maps. Available online: http://maps.stamen.com/#toner/12/37.7706/-122.3782 (accessed on 1 December 2021).
[79] Walch, A.; Castello, R.; Mohajeri, N.; Scartezzini, J.-L. Big data mining for the estimation of hourly rooftop photovoltaic po-tential and its uncertainty. *Appl. Energy* **2020**, *262*, 114404.
[80] House, D.; Lech, M.; Stolar, M. Using Deep Learning to Identify Potential Roof Spaces for Solar Panels. In Proceedings of the 2018 12th International Conference on Signal Processing and Communication Systems (ICSPCS); IEEE: Piscataway, NJ, USA, **2018**; pp. 1–6.
[81] Ruiz, C.A.; Kalkman, I.; Blocken, B. Aerodynamic design optimization of ducted openings through high-rise buildings for wind energy harvesting. *Build. Environ.* **2021**, *202*, 108028, https://doi.org/10.1016/j.buildenv.2021.108028.

INTRODUCTION

	METHOD(S)	CHALLENGE(S)

LITERATURE REVIEW

	METHOD(S)	CHALLENGE(S)	
J1 RQ1	Optimising form-finding parameters in performative computational architecture	■ 17 Form-finding parameters ● 13 Performance aspects ▲ 12 Evolutionary algorithms 3 Swarm algorithms	

TOOL DEVELOPMENT AND PILOT STUDY

	METHOD(S)	CHALLENGE(S)	
J2 RQ2	PART A Developing Optimus tool using self-adaptive ensemble evolutionary algorithm	■ 30 Parameters ▲ 4 Optimisation algorithms	TEST 1
		■ 70 Parameters ▲ 4 Optimisation algorithms	TEST 2
J3 RQ3	PART B Preliminary results of multi-zone approach using pilot high-rise model	■ 100 Parameters ● 2 Daylight metrics ▲ 5 ANN models 1 Optimisation algorithm	TEST 3

METHODOLOGICAL FRAMEWORK

	METHOD(S)	CHALLENGE(S)	
J4 RQ4	PART A Introducing multi-zone optimisation (MUZO) methodology and prediction results of quad-grid and diagrid high-rise scenarios	■ 260 Parameters ● 2 Daylight metrics ▲ 20 ANN models	TEST 4
		■ 220 Parameters ● 2 Daylight metrics ▲ 20 ANN models	TEST 5
J5 RQ5	PART B Optimising high-rise scenarios using the predictive models with Optimus and validation of the MUZO methodology	■ 260 Parameters ● 20 Predictive models ▲ 3 Optimisation algorithms	TEST 4
		■ 220 Parameters ● 20 Predictive models ▲ 3 Optimisation algorithms	TEST 5

CASE STUDY

	METHOD(S)	CHALLENGE(S)
J6 RQ6	Optimising Europoint complex for self-sufficiency in energy consumption and food production using MUZO and Optimus	■ 117 Parameters ● 1 Self-sufficiency in energy 1 Self-sufficiency in food 1 Daylight metric ▲ 45 ANN models 13 Optimisation algorithms

CONCLUSIONS

J Journal
RQ Research question

■ Parameter ● Performance ▲ AI Method

TOC

6 Conclusions

6.1 Introduction

This thesis introduced a computational framework to optimise the self-sufficiency aspects of high-rise buildings during the conceptual phase of the design process. After conducting a systematic review on optimising form-finding parameters in the domain of performative computational architecture (Chapter 2), the Optimus tool to optimise large numbers of decision variables in the architectural design domain (Chapter 3A), and a pilot study having a fundamental machine learning application to test the efficiency of multi-zone approach (Chapter 3B) are presented. Satisfactory results conducted from Chapter 3 led to developing MUZO in Chapter 4. The proposed methodology was tested with two high-rise scenarios with large numbers of design parameters and multiple performance aspects using advanced machine learning techniques with Optimus. Once the proposed methodology was validated, MUZO was utilised to optimise three towers of the Europoint complex in Rotterdam for self-sufficiency in energy consumption and food production. In addition to other optimisation algorithms, the Optimus tool was employed to conduct the results at the building scale in detail and the potential sufficiency at the neighbourhood scale. Despite large numbers of parameters and multiple performance aspects, the proposed MUZO methodology could report well-performing high-rise alternatives in weeks instead of decades. Throughout the research, the relevance of the Optimus tool was proven by outperforming other algorithms owing to self-adaptive strategy and the ensemble of mutation strategies. The architects and engineers can use the output of this research to make design decisions on self-sufficient high-rise buildings during the conceptual design phase in a reasonable timeframe. The thesis is concluded by summarising the contribution of the research, answering the research questions, discussing the limitations with future recommendations.

6.2 Contribution of the research

In architecture and engineering disciplines, there are limitations related to optimising large-scale building design problems considering the performance aspects (with regards to engineering) and form-finding parameters (with regards to architecture) in an integrated design approach. The main contribution of this research is to provide a computational method for dealing with the complexity of integrating architectural and engineering concerns during the conceptual design phase taking into consideration of the self-sufficiency aspects. As one of the inevitable buildings in metropoles, high-rises are a complex type of buildings to design and construct. Based on the review and test results, as well as the results of the case study, the contributions of this research can be summarised as follows:

- A methodological framework for optimising the performance of self-sufficient high-rise buildings considering multiple resources vital for human beings (i.e., energy, food, water).

- Dealing with large-scale high-rise form-finding parameters and self-sufficiency performance aspects that integrates architectural and engineering concerns.

- Predicting computationally expensive performance aspects of self-sufficiency for high-rise buildings with high accuracy regardless of the number of decision variables.

- Providing a decision-making process for designing "especially" in the dense urban environment due to the impact on the performance of the high-rise building in a reasonable timeframe.

- Substantiating the relevance of employing "advanced" optimisation algorithms (such as considering self-adaptive and ensemble of mutation strategies) to cope with the complexity of the design problems.

- Architectural design deals with "tacit knowledge". Comparing different swarm and evolutionary algorithms in architectural design makes the decisions more "**explicit**" and, therefore, "**transparent**".

6.3 Answer to the research questions

6.3.1 Sub-research questions

– **What is the state-of-the-art for optimising form-finding parameters using swarm and evolutionary algorithms in performative computational architecture (Chapter 2)?**

The first sub-question aimed to identify the gaps and needs within the scope of performative computational architecture, which entails form generation, performance evaluation, and optimisation. The review focussed on papers that considered the optimisation of form-finding parameters, which was the fundamental input of form generation, due to their importance in building performance. The results of 100 papers were reported in three categories concerning the performative computational architecture as follows:

The reviewed studies emphasised form-finding parameters related to engineering concerns more than architectural concerns. The reason was that parameters like window-to-wall ratio significantly improved the buildings' energy or daylight performance. However, parameters reflecting engineering concerns were limited in reflecting the architectural concerns. Therefore, an integrated design approach (e.g., as part of the façade design), which could investigate form-finding parameters considering the concerns of both disciplines, was scarce.

For performance aspects, assessment of large-scale design problems, such as high-rise buildings, was limited owing to expensive computation time. Nevertheless, few studies considered a holistic design approach by differentiating the layout and the façade design to decrease the overall complexity. Therefore, holistic approaches in optimising large-scale design problems could be an alternative solution.

The use of optimisation algorithms indicated fundamental versions of the swarm, and evolutionary optimisation algorithms were utilised in most of the reviewed studies. Advanced versions of these algorithms, which consider additional procedures (e.g., self-adaptive parameters, ensembles of mutation strategies), could improve the optimisation performance, thus the performance of the building. Finally, the comparison of different optimisation algorithms with replication of the optimisation

TOC

process was limited in the reviewed papers. Because of the no free lunch theorem in architecture discussed in the review, the convenient optimisation approach in the swarm and evolutionary optimisation would compare different types of algorithms with replication of the optimisation process.

– **What kind of algorithms can discover promising performance results for design problems having large numbers of parameters in the architecture domain (Chapter 3A)?**

The second sub-question investigated the performance of available optimisation algorithms and the proposed optimisation algorithm for design problems having large numbers of parameters in the architecture domain. The Optimus tool was introduced that uses the fundamental version of the differential evolution algorithm with self-adaptive and ensemble procedures. The algorithm in Optimus was compared with a genetic algorithm, particle swarm optimisation, and radial basis function optimisation. Test 1, which considered 30-dimensional benchmark problems developed for the optimisation competitions in Congress on Evolutionary Computation 2005, indicated that Optimus outperformed other algorithms in 19 problems out of 20. Afterwards, Test 2 was focussed on further investigation of the employed algorithms in Test 1, considering a 70-dimensional form-finding design problem. Once again, Optimus outperformed other algorithms by reporting better fitness results at the end of the optimisation process. Based on the evidence conducted from Tests 1 and 2, it was concluded that algorithms with self-adaptive and ensemble procedures could find better performance results when optimising large numbers of design parameters. The reason was that Optimus could adapt the searchability according to the nature of the design problem. In other words, the Optimus tool was able to consider the best rates for crossover and mutation parameters and select the best mutation strategy among different alternatives. On the other hand, other algorithms could not adapt the search strategy since they were using constant values in hyperparameters and singular mutation strategies. Statistical results, optimal solutions, and computation time to complete optimisation processes also supported the arguments regarding the relevance of Optimus to use in design problems having large numbers of parameters, not only in Tests 1 and 2 but also in Tests 4 and 5 (Chapter 4) and the case study (Chapter 5).

- **Can the multi-zone optimisation approach provide better performance results for high-rise buildings in dense urban environments (Chapter 3B)?**

The third sub-question aimed to investigate the impact of the dense surrounding on the performance variances between the ground and sky floor levels of the high-rise buildings. Therefore, Test 3 was formulated in a hypothetical dense urban environment to optimise two conflicting daylight metrics. As one of the most used parameters in the reviewed literature in Chapter 2, overhang length was selected as the form-finding parameter. Additionally, as the visible transmittance of the glazing might have a significant performance enhancement, the glazing type was also considered. The parametric high-rise model was divided into five equal subdivisions (zones) to be considered different design problems to investigate the impact of the dense urban area. Basic application of artificial neural network was utilised to optimise the performance of each separated building subdivision. Results indicated that differentiating overhang length and glazing type for each subdivision could significantly improve the performance of the entire high-rise. The reason was that different parameter values were selected in each subdivision that corresponded to design variations.

- **How can we reach swift and accurate predictions for computationally expensive performance aspects of sustainable buildings in the entire design of high-rises (Chapter 4A)?**

The fourth sub-question focussed on the performance prediction of the entire design of the high-rise building while coping with the computationally expensive simulations. The relevance of the multi-zone approach was addressed in the previous research question. The entire framework of the MUZO methodology was initially introduced, considering the outputs of Chapters 2 and 3. The methodology focussed on developing surrogate models using the dataset collected from the parametric high-rise model in the first two phases. Therefore, swift performance evaluations could be provided to investigate the design problem in the optimisation process, which was the last phase of the MUZO. Two high-rise scenarios with quad-grid and diagrid façade designs were formulated to examine the developed methodology (Tests 4 and 5). The high-rise models were subdivided into ten zones, each to be considered as different design problems. The quad-grid scenario had 26 parameters in each zone corresponding to 260 decision variables in the entire building. The diagrid scenario had 22 parameters in one zone corresponding to 220 decision variables in the entire high-rise model. Form-finding parameters of the models were selected from the ones presented in the literature review (Chapter 2). The thermal property of the glazing was also considered after the potential enhancement observed in Chapter 3B. An advanced artificial neural network application was considered to investigate the parameters of surrogate modelling with 10-fold cross-validation. Results indicated that each prediction of each performance aspect in every zone required to use of a different set of hyperparameters to provide the highest prediction accuracy. The relevance of the results was also validated by comparing the prediction accuracies of similar works in the same domain. Although each predictive model had different numbers of design parameters (that suggested various design complexities), the surrogate modelling phase of MUZO could find promising prediction accuracies.

– **How can we optimise large numbers of design parameters while investigating multiple performance aspects of sustainable high-rise alternatives (Chapter 4B)?**

The fifth sub-question aimed to investigate the optimisation of high-rise buildings considering large numbers of design parameters and multiple performance aspects. Although the previous sub-question answered how to achieve swift and accurate performance predictions using artificial neural networks for the entire high-rise design, optimising the large numbers of design parameters remained a challenge. The third phase of the MUZO methodology focussed on optimisation and decision-making using the predictive models developed in the second phase. As indicated in the literature review (Chapter 2), high awareness of the search space could be achieved only by employing several optimisation algorithms and replicating the optimisation processes. Therefore, the optimisation phase of the MUZO methodology entails the use of multiple optimisation algorithms with replications. Optimisation results conducted from Tests 4 and 5 indicated that search behaviours and results of the optimisation algorithms presented variances for both high-rise scenarios. Even though the Optimus tool found the best performance score in most optimisation problems, other algorithms could find slightly better results in several optimisation models. In this competitive comparison, the performance of the Optimus emerged as the most promising algorithm, not only because of finding better results but also discovering optimal solutions with fewer deviations and less computation time. The relevance of the MUZO in optimising large numbers of design parameters was also proven by comparing the high-rise buildings having the same set of parameter values for the entire building instead of differentiating them as in the proposed methodology. Although optimising large numbers of design parameters with a self-adaptive ensemble algorithm caused promising design solutions, this was possible because of the predictive models. Otherwise, more than 17 years were required to complete the computations to gain the same awareness of the search space using simulation models instead of predictions.

- **What is the potential of the developed computational method for self-sufficiency in energy consumption and food production at the building and neighbourhood scales (Chapter 5)?**

The final sub-question focussed on investigating the potentials of an existing high-rise building as a case study, using the MUZO methodology and Optimus tool. The Europoint complex in Rotterdam, the Netherlands, consists of three high-rise buildings, was focussed on optimising two self-sufficiency aspects: energy consumption and food production. A building skin was proposed to optimise the façade design, separately for each tower. Form-finding parameters related to the building's façade in Chapter 2 and thermal and transmittance properties of the glazing were considered as design parameters. The buildings were examined in detail for self-sufficiency at the building scale. The developed model was also investigated to address the potential at the neighbourhood scale. In addition to self-sufficiency aspects, daylight availability was considered in the optimisation as a constraint. The design problem was optimised with 13 algorithms, including the Optimus tool. Results indicated that 100% food sufficiency for lettuce crops could be reached at the building scale for 1800 residents and the neighbourhood scale up to 27,000 people. On the other hand, 43.7% energy sufficiency was reached using building integrated photovoltaic panels at the building scale. Self-sufficiency in energy varied between 47.8% and 11.6% at the neighbourhood scale because of the energy demand of the vertical farms. Optimisation results suggested different design decisions for each tower. Once more, the MUZO methodology enhanced the building performance up to 21% compared to regular high-rise scenarios using the same set of parameters for the entire design of the buildings. Ultimately, the Optimus tool successfully coped with 117-dimensional design problems in both scales by discovering the promising self-sufficiency scores in the least computation time.

6.3.2 Main research question

- **How can we optimise the performance of self-sufficient high-rise buildings using artificial intelligence in the conceptual design phase?**

The main question was addressed to investigate achieving well-performing high-rise alternatives while integrating performance aspects of self-sufficiency in the conceptual design phase. The literature review showed that the simulation-based optimisation processes would limit the investigation of well-performing alternatives because of the expensive computational time. The literature review also indicated that using a single optimisation algorithm might lead to discovering nonoptimal solutions because of the no free lunch theorem. Therefore, a novel computational framework (MUZO) was proposed in this research.

The MUZO aimed to use artificial intelligence methods to make design decisions on self-sufficient performance aspects with high awareness of the search space in the conceptual phase. However, considering several computational steps was insufficient because the optimisation problem was also challenging due to the large number of parameters involved in the decision-making. Therefore, the Optimus tool was introduced by validating its efficiency with Tests 1 and 2.

The relevance of the MUZO using the Optimus tool was validated with Tests 3, 4, and 5. Finally, the self-sufficiency results of the case study were conducted at the building and neighbourhood scales. The tests and the case study suggested that the MUZO methodology with the Optimus tool could achieve remarkable performance solutions while optimising high-rise buildings for self-sufficiency. In this way, the use of the MUZO with Optimus tool can support optimising the performance of self-sufficient high-rise buildings by providing:

- Swift and accurate performance predictions in multiple self-sufficiency criteria,
- High awareness of the search space with large numbers of design parameters,
- Conceptual design decisions for well-performing self-sufficient high-rises in a reasonable timeframe.

6.4 Limitation of the research

This dissertation proposed an innovative computational framework to optimise the performance of self-sufficient high-rise buildings using artificial intelligence. Despite the examples focusing on only energy-related performance aspects, the novelty of this study was to cope with multiple performance aspects for self-sufficiency in multiple resources through the power of computational tools and methods. Additionally, artificial intelligence methods in the computational framework made it possible to investigate such complex design tasks in an acceptable amount of time during the conceptual design phase. Nevertheless, the proposed framework has computational limits if the study is extended to larger self-sufficiency scales, periods, parameters, and performance criteria.

The research dealt with considerable data for investigating self-sufficiency in building scale. A computational challenge remains to cope with the neighbourhood and city-scale data. Investigating self-sufficiency on bigger scales may require involving other artificial intelligence methods. The case study of the research in Chapter 5 addressed the potential sufficiency results at the neighbourhood scale. Nevertheless, investigation of this scale had a limitation owing to considering only the case buildings to allocate renewable resources instead of involving the other buildings in the surrounding. The climate conditions might not provide the same efficiency of resource usage, such as the variance of solar radiation between the summer and winter periods. The proposed computational framework can deliver various self-sufficiency scores for different periods. The extension of the models to predict self-sufficiency for smaller periods, such as monthly or daily self-sufficient instead of annual predictions, also remained the research's limitation.

Specifically, the research dealt with the challenges during the conceptual design owing to the importance of decisions given in the early phases of the self-sufficient high-rise buildings. Most of the form-finding variables reported in the literature review were used in Tests 3, 4, and 5 that were based on hypothetical scenarios. However, the same flexibility could not be considered in the case study because of focusing on a real-world scenario. Finally, the developed computational framework demonstrated self-sufficiency only for energy and food via photovoltaic panels and plant factories. Although daylight availability was defined as a constraint in the optimisation problem of the case study, the involvement of additional resources to achieve higher self-sufficiency scores remained a research limitation.

6.5 Future recommendations

Although the relevance of the proposed computational framework and the optimisation tool was proven with various tests and a case study, extension of the research into the experimental study will be beneficial to validate the collected simulation results from the digital environment with the monitored data. Additionally, several recommendations can be mentioned for each category of the self-sufficiency framework, explained in Chapter 1, as follows:

Scale of the self-sufficiency

Although the potential of self-sufficiency at the neighbourhood scale was addressed in the case study, a deep investigation, which considers the other buildings in the surrounding, is necessary. In this way, not only placing the photovoltaic panels on the rooftops but also other potentials of the surrounding (e.g., locating wind miles, distributing the various crops of the vertical farms in different locations, considering unused parts of the neighbourhood for urban farming) could be used to achieve higher self-sufficiency scores in the neighbourhood scale. Despite the increase in complexity because of the scale, MUZO methodology can be applicable since collecting data from various locations is possible. Achieving promising self-sufficiency scores at the neighbourhood scale can assist in focusing on the city scale as the next challenge.

Period of the self-sufficiency

The data collected in this research was based on annual results of the performance aspects. The extension of the research in smaller periods such as monthly self-sufficient or weekly self-sufficient can be beneficial to optimise the insufficient periods of the model. However, the smaller period to be considered in the simulations will increase the size of the data; thus, the required computation time to complete the evaluation. Therefore, the usage of the GPU of the computer or a computer having very high computation power can be necessary for data preparation and the development of surrogate modelling.

Parameters of the self-sufficiency

Hypothetical scenarios considered most of the form-finding parameters related to the shape and façade of the building reported in Chapter 2. However, the case study could only use the form-finding parameters related to façade design due to the high-rise buildings' existing layout. Hence, building shape-related form-finding parameters can improve the self-sufficiency scores in a real-world high-rise scenario. Focusing on other parameters of self-sufficiency (i.e., related to the building construction and building operation) may increase the complexity of the design problem. Therefore, considering separate optimisation problems for conceptual, construction, and operation phases can be beneficial to cope with the large numbers of parameters belonging to various phases of the building.

Performance aspects of the self-sufficiency

Promising self-sufficiency results at the building scale was achieved in energy considering the façade skin and rooftop to place photovoltaic panels. There is still room to integrate various systems for energy that may enhance the self-sufficiency score, such as wind miles and ducts, thermal storage, battery systems. In the neighbourhood scale, the potentials can be identified using convolutional neural networks to find the optimal place for renewable resources. Additionally, a detailed setup of the HVAC system (using other plug-ins in Grasshopper 3d for condenser or the VRF loops) may also improve energy efficiency in future works. Regarding the food, only the sufficiency of lettuce crops was demonstrated using a closed farming system. The simulation models can be used to evaluate the energy demand of the other crops and integrate the design parameters of the closed farming systems to optimise the energy consumption. In the neighbourhood, instead of growing agricultural products in one location, distributions of the various crops in multiple locations using closed and urban farming setups can be investigated to find the best food supply with the least energy demand and CO_2 emissions. The impact of the occupancy schedules of various building programs can be studied that may suggest a more complex simulation model resulting in higher computation time. Even though closed farming systems provide significant efficiency in water consumption, systems and design parameters related to self-sufficiency in water can be addressed in terms of harvesting and recycling. Finally, the economic assessment of the proposed self-sufficient high-rise can take place to provide profitable alternatives.

Curriculum Vitae

Berk Ekici

DoB: 18.09.1988
PoB: İzmir – Turkey
Skype: berk.ekici
E-mail: berkekici.com@gmail.com
https://www.linkedin.com/in/berkekici/
https://www.researchgate.net/profile/Berk-Ekici

Education and Training

08.2016 – 06.2022	**Doctor of Philosophy** Delft University of Technology, The Netherlands *Towards Self-Sufficient High-Rises:* *Performance Optimisation using Artificial Intelligence*
09.2011 – 11.2014	**Master of Science in Architecture** Yaşar University, Turkey *Multi-Performance Based Computational Design in* *Architecture: Integrated High-Rise Buildings*
01.2013 – 07.2013	**Visiting Scholar (Erasmus)** Universidad Politecnica de Valencia, Spain
09.2006 – 07.2011	**Bachelor of Architecture** Yaşar University, Turkey
06.2010 – 09.2010	**Internship (Erasmus)** Michele Gambato Architetto, Italy

07.2009 – 08.2009 **Internship**
Beset Architectural Studio Construction Inc., Turkey

08.2008 – 10.2008 **Internship**
Y Architecture Services, Turkey

Work Experience

09.2019 – 07.2021 **Adjunct Instructor**
Delft University of Technology, The Netherlands

02.2019 – 06.2019 **Adjunct Instructor**
Yaşar University, Turkey

10.2011 – 01.2019 **Research Assistant**
Yaşar University, Turkey

01.2010 – 06.2010 **Student Assistant**
Yaşar University, Turkey

Contributed Research Projects

12.2017 – 06.2019 **Terrestrial Laser Scanning Technologies in Cultural Heritage Documentation**
Yaşar University, Turkey (Funded by YU Research Centre)

09.2016 – 06.2017 **Designing Asymmetric Shaped Shell System with Aerated Concrete Blocks using Computational Design Techniques**
Yaşar University, Turkey (Funded by YU Research Centre)

09.2016 – 06.2017 **An Approach for Energy Efficient Retrofit of Architectural Studio Building in University Campus İzmir (Project no: 114M803)**
Yaşar University, Turkey (Funded by Tubitak)

Supervision

2020	**Humble Giants: Computational Intelligence for Designing More Sustainable High-Rise Buildings** Delft University of Technology (MSc Student: Fredy F. Mora)
2018	**Energy and Daylight Performance Optimization for High-Rise Office Buildings** Yaşar University (MSc Student: Muhittin Yufka)

Academic Organisations and Activities

Contributing as Invited Reviewer

2018 - Present	Simulation: Tran. of the Soc. for Mod. & Sim. Int. (Sage) Architectural Eng. and Design Mngmt. (Taylor&Francis) Building Research and Information (Taylor&Francis) Int. Journal of Ambient Energy (Taylor&Francis) Journal of Façade Des. and Eng. (TU Delft open) Automation in Construction (Elsevier) Building and Environment (Elsevier) Energy and Buildings (Elsevier) Solar Energy (Elsevier) Structures (Elsevier) Buildings (MDPI)

Organisations

2016	Technical committee member, IEEE congress on computational intelligence, Special session no 36: Evolutionary computation in architectural design Vancouver, Canada
2015	Technical committee member, IEEE congress on evolutionary computation, Special session no 41: Evolutionary computation in architectural design Sendai, Japan

2012	International symposium organisation, 2012 computational design international symposium and workshop (Contributors: Yaşar University, TU Delft, University of Salford, Polytechnic of Turin, İzmir University of Economics) İzmir, Turkey

Honours, Awards, Grants

2018	Belgrade, Serbia Travel grant, COST action 1403 adaptive façade network
2018	Las Palmas, Spain Best paper award, 12th international symposium on TMCE
2012	Uşak, Turkey 3rd honourable mention, Bus terminal complex competition
2011	Manisa, Turkey 4th honourable mention, Municipality building competition
2011	Yaşar University, Turkey Top student of the department
2011	Chamber of Architects Top student of the department (in İzmir branch)
2011	Yaşar University, Turkey The best project award
2011	Yaşar University, Turkey Dean’s success award
2008-2011	Yaşar University, Turkey High honour award (4 times)

List of Publications

This study

Journal Articles

1) **B. Ekici**, O. F. S. F. Turkcan, M. Turrin, I. S. Sariyildiz, M. F. Tasgetiren, "*Optimising High-Rise Buildings for Self-Sufficiency in Energy Consumption and Food Production Using Artificial Intelligence: Case of Europoint Complex in Rotterdam*", 2022, Energies, 15(2): 660.

2) **B. Ekici**, Z. T. Kazanasmaz, M. Turrin, M. F. Tasgetiren, I. S. Sariyildiz, "*Multi-zone optimisation of high-rise buildings using artificial intelligence for sustainable metropolises. Part 1: Background, methodology, setup, and machine learning results*", 2021, Solar Energy, 224: 373-389.

3) **B. Ekici**, Z. T. Kazanasmaz, M. Turrin, M. F. Tasgetiren, I. S. Sariyildiz, "*Multi-zone optimisation of high-rise buildings using artificial intelligence for sustainable metropolises. Part 2: Optimisation problems, algorithms, results, and method validation*", 2021, Solar Energy, 224: 309-326.

4) C. Cubukcuoglu, **B. Ekici**, M. F. Tasgetiren, I. S. Sariyildiz, "*OPTIMUS: Self-adaptive differential evolution with ensemble of mutation strategies for grasshopper algorithmic modeling*", Algorithms, 2019, 12(7): 141.

5) **B. Ekici**, C. Cubukcuoglu, M. Turrin, I. S. Sariyildiz, "*Performative computational architecture using swarm and evolutionary optimisation: A review*", Building & Environment, 2019, 147: 356-371.

Conference Proceedings

1) **B. Ekici,** Z. T. Kazanasmaz, M. Turrin, M. F. Tasgetiren, I. S. Sariyildiz, "*A Methodology for daylight optimisation of high-rise buildings in the dense urban district using overhang length and glazing type variables with surrogate modelling*", 2019, Journal of Physics: Conference Series 1343 (1), 012133 (Poster presentation).

Datasets

1) **B. Ekici**, O. F. S. F. Turkcan, M. Turrin, I. S. Sariyildiz, M. F. Tasgetiren, “Multi-zone simulation results of Europoint complex for self-sufficiency in energy consumption and food production in Rotterdam”, 2021, https://doi.org/10.4121/17129420.v1, 4.TU.ResearchData.

2) **B. Ekici**, Z. T. Kazanasmaz, M. Turrin, M. F. Tasgetiren, I. S. Sariyildiz, “Multi-zone simulation results on ASE and sDA daylight metrics for parametric high-rise model with quad grid and diagrid facade in a highly dense hypothetical urban district using dry summer climate weather data”, 2020, https://doi.org/10.4121/uuid:8538ac2f-3a78-4923-8fca-5beb50017299, 4.TU.ResearchData.

Others

Journal Articles

1) A. Kirimtat, O. Krejcar, **B. Ekici**, M. F. Tasgetiren, “*Multi-objective energy and daylight optimization of amorphous shading devices in buildings*”, 2019, Solar Energy, 185: 100-111

2) B. Kundakci-Koyunbaba, **B. Ekici**, A. Kirimtat, I. Chatzikonstantinou, A. Cilasun-Kunduraci, E. Yildirim, “*Energy-efficient retrofit proposal for an architectural studio building with frame structure: A case study*”, (work in progress)

Book Chapters

1) C. Cubukcuoglu, A. Kirimtat, **B. Ekici**, M. F. Tasgetiren, “*Evolutionary algorithms for theatre hall acoustics*”, 2019, Optimization in industry - Present practices and future scopes, Pp: 55- 83, Springer.

2) E. Cevizci, S. Kutucu, M. Morales-Beltran, **B. Ekici**, M. F. Tasgetiren, “*Structural optimization for masonry shell design using multi-objective evolutionary algorithms*”, 2019, Optimization in industry - Present practices and future scopes, Pp:85-119, Springer.

3) A. Kirimtat, **B. Ekici**, C. Cubukcuoglu, M. F. Tasgetiren, I. S. Sariyildiz, “*Evolutionary algorithms for designing self-sufficient floating neighborhoods*”, 2019, Optimization in industry - Present practices and future scopes, Pp: 121-147, Springer.

Conference Proceedings

1) N. P. Unlu, **B. Ekici**, I. Chatzikonstantinou, I. S. Sariyildiz, M. F. Tasgetiren, C. Cubukcuoglu, "*Diagrid façade design for public pool building using differential evolution*", 2018, Twelfth international tools and methods of competitive engineering symposium, Las Palmas de Gran Canaria, 265-274. (Full paper presentation)

2) E. Yildirim, **B. Ekici**, C. Cubukcuoglu, I. Chatzikonstantinou, I. S. Sariyildiz, "*Optimization of free form long span roof structure for pool facility using evolutionary algorithms*", 2017, International symposium for production research, Wien, 891-902. (Full paper presentation)

3) E. Kurtbas, C. Cubukcuoglu, **B. Ekici**, A. Cilasun Kunduraci, I. Kahraman, "*Addressing a façade design for healthcare facility using multi-objective optimization*", 2017, International symposium for production research, Wien, 481-494. (Full paper presentation)

4) M. Paldrak, C. Cubukcuoglu, **B. Ekici**, M. F. Tasgetiren, A. Cilasun Kunduraci, "*Diagrid and honeycomb façade design optimization with multi-objective evolutionary algorithms*", 2017, International symposium for production research, Wien, 555-567. (Full paper presentation)

5) G. Yavuzarslan, **B. Ekici**, C. Cubukcuoglu, A. Cilasun Kunduraci, B. Kundakci Koyunbaba, "*Envelope design of healthcare public space using multi-objective optimization*", 2017, International symposium for production research, Wien, 867-880. (Full paper presentation)

6) N. P. Unlu, M. F. Tasgetiren, **B. Ekici**, C. Cubukcuoglu, I. Chatzikonstantinou, "*Structure optimization of shelter design for semi-closed public space in urban*", 2017, International symposium for production research, Wien, 811-823. (Full paper presentation)

7) F. Ozbey, C. Cubukcuoglu, **B. Ekici**, A. Cilasun Kunduraci, I. Chatzikonstantinou, "*Bi-objective visual perception optimization for a healthcare education facility*", 2017, International symposium for production research, Wien, 511-520. (Full paper presentation)

8) A. O. Gorgun, C. Cubukcuoglu, **B. Ekici**, B. Kundakci Koyunbaba, I. Kahraman, "*Diagrid façade design using multi-objective evolutionary algorithms*", 2017, International symposium for production research, Wien, 331-344. (Full paper presentation)

9) B. Yildiz, **B. Ekici**, I. Chatzikonstantinou, S. Kutucu, I. S. Sariyildiz, "*Visual perception based multi- objective optimization to housing design in urban context*", 2017, International symposium for production research, Wien, 903-914. (Full paper presentation)

10) S. Karaman, **B. Ekici**, C. Cubukcuoglu, B. Kundakci Koyunbaba, I. Kahraman, "*Design of rectangular façade modules through computational intelligence: Case of common space in healthcare building*", 2017, IEEE congress on evolutionary computation, San Sebastian, 1021-1028. (Full paper presentation)

11) M. Yufka, **B. Ekici**, C. Cubukcuoglu, I. Chatzikonstantinou, I. S. Sariyildiz, "*Multi-objective skylight optimization for a healthcare facility foyer space*", 2017, IEEE congress on evolutionary computation, San Sebastian, 1008-1014. (Full paper presentation)

12) **B. Ekici**, I. Chatzikonstantinou, I. S. Sariyildiz, M. F. Tasgetiren, Q. K. Pan, "*A Multi-objective self- adaptive differential evolution algorithm for conceptual high-rise building design*", 2016, IEEE World congress on computational intelligence, Vancouver, 2272-2279. (Full paper presentation)

13) C. Cubukcuoglu, I. Chatzikonstantinou, **B. Ekici**, I. S. Sariyildiz, M. F. Tasgetiren, "*Multi-objective optimization through differential evolution for restaurant design*", 2016, IEEE World congress on computational intelligence, Vancouver, 2288-2295. (Full paper presentation)

14) E. E. Aydin, O. Dursun, I. Chatzikonstantinou, **B. Ekici**, "*Optimisation of energy consumption and daylighting using building performance surrogate model*", 2015, 49th international conference of the architectural science association, Melbourne, 536-546. (Full paper presentation)

15) I. Chatzikonstantinou, **B. Ekici**, B. Kundakci Koyunbaba, I. S. Sariyildiz, "*Multi-objective diagrid façade optimization using differential evolution*", 2015, IEEE congress on evolutionary computation, Sendai, 2311-2318. (Full paper presentation)

16) **B. Ekici**, S. Kutucu, I. S. Sariyildiz, M. F. Tasgetiren, "*Addressing the high-rise form finding problem by evolutionary computation*", 2015, IEEE congress on evolutionary computation, Sendai, 2253- 2260. (Full paper presentation)

17) O. Dursun, **B. Ekici**, I. S. Sariyildiz, "*Time-cost optimization at the conceptual design stage using differential evolution: Case of single-family housing projects in Germany*", 2015, IEEE congress on evolutionary computation, Sendai, 2237-2244. (Full paper presentation)

18) **B. Ekici** and S. Kutucu, "*Bilgisayar destekli kavramsal tasarım yaklaşımı ve çoklu performansa dayalı tümleşik yüksek yapılar (Computer aided conceptual design approach and multi-performance based integrated high-rise buildings)*", 2014, 8. mimarlıkta sayısal tasarım ulusal sempozyumu, Izmir, 85-94. (Full paper presentation)

19) **B. Ekici** and E. Özmehmet, "*Çok yönlü bina enerji yazılım araçlarının analizi (Analysis of multi-purpose building energy software tools*)", 2012, Sürdürülebilir yapı tasarımı kongresi bildiriler kitabı, Izmir, 163-176. (Full paper presentation)

22#10

Towards Self-Sufficient High-Rises

Performance Optimisation using Artificial Intelligence

Berk Ekici

Population growth and urbanisation trends bring many consequences related to the increase in global energy consumption, CO_2 emissions and a decrease in arable land per person. High-rises have been one of the inevitable buildings of metropoles to provide extra floor space since the early examples in the 19^{th} century. Therefore, optimisation of high-rise buildings has been the focus of researchers because of significant performance enhancement, mainly in energy consumption and generation. Based on the facts of the 21^{st} century, optimising high-rise buildings for multiple vital resources (such as energy, food, and water) is necessary for a sustainable future.

This research suggests "*self-sufficient high-rise buildings*" that can generate and efficiently consume vital resources in addition to dense habitation for sustainable living in metropoles. The complexity of self-sufficient high-rise building optimisation is more challenging than optimising regular high-rises that have not been addressed in the literature. The main challenge behind the research is the integration of multiple performance aspects of self-sufficiency related to the vital resources of human beings (energy, food, and water) and consideration of large numbers of design parameters related to these multiple performance aspects. Therefore, the dissertation presents a framework for performance optimisation of self-sufficient high-rise buildings using artificial intelligence focusing on the conceptual phase of the design process. The output of this dissertation supports decision-makers to suggest well-performing high-rise buildings involving the aspects of self-sufficiency in a reasonable timeframe.

A+BE | Architecture and the Built Environment | TU Delft BK