Case Studies in Construction Materials xxx (2018) xxx–xxx

Contents lists available at ScienceDirect

Case Studies in Construction Materials

journal homepage: www.elsevier.com/locate/cscm

Case study

Using algorithms to designate pre-fabricated wall materials: A case study with two implementation methods

Shenghuan Zhao

Department of Architecture, Built Environment and Construction Engineering, Politecnico di Milano, Piazza Leonardo da Vinci, 32, 20133, Milano, Italy

ARTICLE INFO

Article history:
Received 24 October 2018
Received in revised form 17 January 2019
Accepted 18 January 2019

Keywords:
Optimization
Design automation
Decision making
Design science
Fabrication

ABSTRACT

Designating wall materials manually is neither efficient nor competent to find the ideal solution. Therefore, the current study aims to develop a new designation approach based on multi-objective optimization which can automatically explore a larger space of solutions and obtain the optimal material combination. The core concept is to use wall properties as objectives to search for satisfying combinations, rather than the thresholds to validate manually proposed solutions. A case study is carried out to find material combinations with favourable U value, specific heat capacity, and the total material price. It is realised by two different research methods: using the k-nearest neighbours (k-NN) algorithm in the Python environment, and using the Strength Pareto Evolutionary Algorithm 2 (SPEA2) in the Grasshopper. Finally, the usability test was conducted among 15 architects. Test results confirmed that this new approach could save much time and find solutions with better performance. The present study has significance in reducing architects' repetitive work while speeding up the decision-making process.

1. Introduction

Designating wall material is still done using an old fashioned method: architects look up in the material handbook, select materials, give the thickness of the different layers, and then check the wall properties by using Excel, or local software like BRE's U value calculator, or other online thermal calculators [1–3]. Designers input the wall material information on these web pages and they automatically output the whole scenario of wall properties, like the R value, U value, and the moisture condition. If the results of wall performance do not satisfy construction codes or clients' requirements, architects have to change materials or thicknesses and iterate this process until the compliance goal is reached. Due to too many trade-offs among objectives, this kind of human-brain-based method is neither efficient nor able to find the ideal solution. For example, finding the embodied energy and total price rise when building materials save energy in the operation stage. To find the solution balancing these design aims is a multi-objective optimization problem, which is impossible to be manually handled by architects.

There have been several other investigations using algorithms to design the building envelope. Azari et al. [4] used an artificial neural network (ANN) and genetic algorithms to select envelope materials, concerning the building energy consumption and the life cycle analysis. Castro-Lacouture et al. [5] proposed an integer optimization model to select building materials for achieving more credits in the LEED rating system, which provides a framework to create cost-saving green buildings. Zavadskas et al. [6] set up a model with the MULTIMOORA method to select materials for residential house elements.

Nevertheless, drawbacks and gaps exist between precedent investigations and the wall composition designation in reality. First of all, material optimization in the literature has usually been conducted as an affiliation to the optimization of

https://doi.org/10.1016/j.cscm.2019.e00220

building form or other issues, with few material parameters correlated. However, the material designation is already a complicated and systematic problem by itself, and it is regularly separated from finding the best window-wall-ratio or other design problems in real design projects. Second, LEED credits and some other objectives pursued in these studies are not actually compulsory for every project. Moreover, design decisions of wall material designation are made by architects in real life, but precedent studies were mainly conducted by engineers using methodologies which were not only too time-consuming but also complicated for architects. In conclusion, the practical applicability of prior research is limited.

Accordingly, the current research tries to automate the material designation process by more necessary and representative objectives, which will be illustrated in chapter 2.2. The calculation model built herein can be applied to any new projects and conditions. Meanwhile, methodologies in the current study will be more understandable and accessible to common architects. A pre-fabricated wall typology with three layers will be illustrated as a case study. It will be realised with two different methods: the k-NN algorithm in a programming environment of Python and an Evolutionary Algorithm in the Grasshopper [7].

2. Basic information

2.1. Background

The case study herein is a prefabricated wall composed of three layers: Extruded Cement Panel (ECP), an insulation layer, and Autoclaved Lightweight Concrete (ALC) panel (Fig. 1). This kind of wall composition typology has been utilized in many construction projects over East Asia, especially in Japan and China, such as the project of the new Beijing municipal government office building. Designing one traditional wall layer includes two aspects: selecting the material and deciding its thickness. When it comes to the prefabricated wall, material types and their thicknesses are usually arranged in pairs that make the optimization problem much easier. A short introduction of three layers will be given in the following paragraphs.

The exterior layer is the extruded cement panel (ECP), is a kind of hollow cement strip. It is made from waste concrete powder [8] or Portland cement, fibres (pulp fibre, polypropylene fibre, etc.), and siliceous materials (quartz powder). After the extrusion moulding under vacuum and high pressure, intermediate products go through a low-temperature steam curing and another high-temperature with high-pressure curing to turn into the final product. The whole manufacturing process does not produce wastewater, gas, and hazardous chemicals. This case study offers three types of ECP with different thicknesses (Table 1), in an interval of 10 mm.

Only three kinds of insulation materials are provided but with 19 various thicknesses (Table 2). Expanded polystyrene (EPS) and extruded polystyrene (XPS) board are the most popular insulation materials in construction projects [9]. On top of them, Polyurethane (PU) board is an alternative here. The thickness interval of XPS is 5 mm while the other two kinds have an interval of 10 mm.

Autoclaved Lightweight Concrete (ALC) panel which has the strong compressive strength [10] and lateral stiffness [11] is the inner base of the wall. Due to the ALC board used for an external wall must be thicker than 150 mm, only seven kinds of thickness were found in the market. As a result, there are 3 kinds of ECP, 17 types of the insulation layer, 7 sorts of ALC, and 399 possible combinations in total. More alternatives can be brought about by adding material kinds and changing their properties.

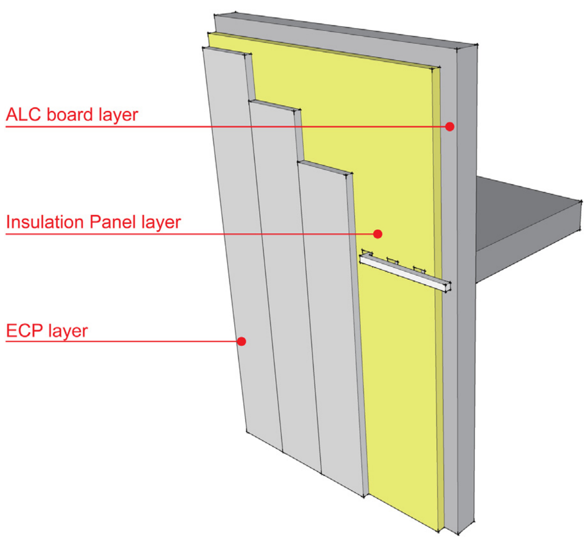

Fig. 1. The prefabricated wall used for the case study.

Tables 1–3 shows the detailed information of these three layers. R_n, C_{Pn}, m_n, P_n are four variables participating in the optimization. The D is the thickness of each layer, λ is the conductivity of each material. According to these two indicators, we can get the thermal resistance R_n of the layer n. C_{Pn} is the heat capacity of layer n, m_n is the areal density of layer n. The price information is obtained from the largest e-commerce website Alibaba.

2.2. Design mission and fitness functions

Due to the relative complexity and difficulty of high-dimensional optimization [12], the present study takes three objectives to do the optimization. Speaking of specific objectives, there are many criteria for the wall material selection: building energy consumption, embodied energy, the mould risk [13], constructability, etc. Which objectives are more necessary and urgent? Among them, the energy efficiency, strongly dependent on the thermal behaviour of walls, is the most strengthened aim in construction codes all over the world. It is the reason why there are so many wall thermal calculators in the market, which have been introduced in the first chapter. Because the wall thermal behaviour is typically decided by both thermal resistance and heat capacity [14], total U value (thermal transmittance) and the total heat capacity are two main objectives pursued in the current study. Meanwhile, the full price is chosen to make the current study closer to real design projects. In conclusion, the design mission herein is to find the best one from 399 solutions of material combination, with regard to three performance objectives: total thermal transmittance, total specific heat capacity, and full price. Detailed information will be discussed in the following paragraphs.

The U value was first introduced to lessen internal condensation and then changed mainly to conserve the fuel and power [15]. Smaller U value can save more energy. For commercial usage, the U value calculations are made according to ISO

Table 1
Information of ECP layer.

Names	d (mm)	λ_1(W/m·K)	R_1(m^2·k/W)	C_{P1}(kj/kg·K)	ρ_1(kg/m^3)	m_1(kg/m^2)	P_1(CN¥/m^2)
A1	40	0.53	0.08	1.05	75	3	350
A2	50	0.53	0.09	1.05	75	3.75	380
A3	60	0.53	0.11	1.05	75	4.50	400

Table 2
Information of insulation layer (B1-B7: XPS, B8-B11: EPS, B13-B19: PU).

Names	D (mm)	λ_2(W/m·K)	R_2(m^2·k/W)	C_{P2}(kj/kg·K)	ρ_2(kg/m^3)	m_2(kg/m^2)	P_2(CN¥/m^2)
B1	20	0.03	0.67	5.35	25	0.50	9.60
B2	25	0.03	0.83	5.35	25	0.63	12
B3	30	0.03	1	5.35	25	0.75	14.40
B4	35	0.03	1.17	5.35	25	0.88	16.80
B5	40	0.03	1.33	5.35	25	1	19.20
B6	45	0.03	1.50	5.35	25	1.13	21.60
B7	50	0.03	1.67	5.35	25	1.25	24
B8	30	0.04	0.75	2.41	18	0.54	12
B9	40	0.04	1	2.41	18	0.72	16
B10	50	0.04	1.25	2.41	18	0.90	20
B11	60	0.04	1.50	2.41	18	1.08	24
B12	70	0.04	1.75	2.41	18	1.26	28
B13	20	0.02	1	4.77	30	0.60	11.20
B14	25	0.02	1.25	4.77	30	0.75	14
B15	30	0.02	1.50	4.77	30	0.90	16.80
B16	35	0.02	1.75	4.77	30	1.05	19.60
B17	40	0.02	2	4.77	30	1.20	22.40
B18	45	0.02	2.25	4.77	30	1.35	25.20
B19	50	0.02	2.50	4.77	30	1.50	28

Table 3
Information of ALC Layer.

Names	D (mm)	λ_3(W/m·K)	R_3(m^2·k/W)	C_{P3}(kj/kg·K)	ρ_3(kg/m^3)	m_3(kg/m^2)	P_3(CN¥/m^2)
C1	150	0.16	0.93	1.13	520	78	75
C2	175	0.16	1.09	1.13	520	91	87.50
C3	200	0.16	1.25	1.13	520	104	100
C4	225	0.16	1.41	1.13	520	117	112.50
C5	250	0.16	1.56	1.13	520	130	125
C6	275	0.16	1.72	1.13	520	143	137.50
C7	300	0.16	1.88	1.13	520	156	150

6946:2017, as function 1 and function 2 illustrated. R is the total resistance of the whole wall while R_1 to R_3 is the thermal resistances of three layers respectively. R_i and R_e stand for the internal and external surface thermal resistances. For a vertical wall exposing to the air, R_i is 0.11 while R_e is 0.14.

$$R = R_1 + R_2 + R_3 \tag{1}$$

$$U = \frac{1}{R_i + R + R_e} \tag{2}$$

Heat capacity (thermal capacity) [J/(kg*K)] is a measurable physical quantity. It equals the ratio of the heat added to an object to the resulting temperature change. Therefore, a wall has more moderate temperature variations if it has a higher heat capacity. The total heat capacity of a wall can be calculated according to the Functions 3 and 4.

$$C_{Pn} = m_n \times C_n^n \tag{3}$$

$$C_{P-total} = \left(\frac{m_1}{m_{total}}\right) C_{P1} + \left(\frac{m_2}{m_{total}}\right) C_{P2} + \left(\frac{m_3}{m_{total}}\right) C_{P3} \tag{4}$$

The total cost of the wall is a sum of all three layers' prices, in the unit of Chinese Yuan per square meter. Apparently, the lower the wall price, the better. In conclusion, we should minimize the U value (Function 5) and the financial cost (Function 7), while maximizing the heat capacity (Function 6).

$$U = \frac{1}{0.25 + R_1 + R_2 + R_3} \tag{5}$$

$$C_{P-total} = \left(\frac{m_1}{m_1 + m_2 + m_3}\right) C_{P1} + \left(\frac{m_2}{m_1 + m_2 + m_3}\right) C_{P2} + \left(\frac{m_3}{m_1 + m_2 + m_3}\right) C_{P3} \tag{6}$$

$$P_{total} = P_1 + P_2 + P_3 \tag{7}$$

3. Implementation in Spyder with the KNN algorithm

3.1. Spyder and KNN algorithm

Python is a popular coding language with a broad application, so its basic information will not be repeated herein. Spyder is an open source platform for Python programming (Fig. 2), offering a remarkable integration of editing, analysis,

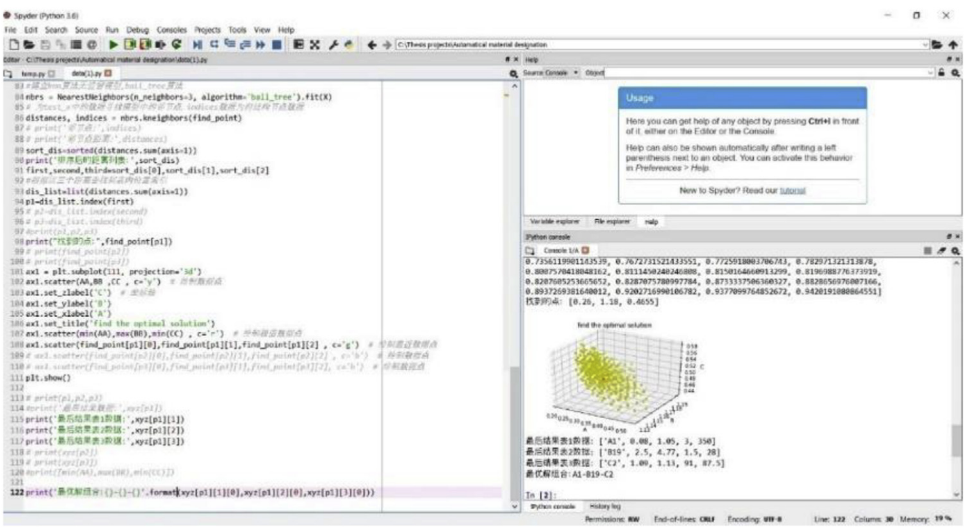

Fig. 2. The interface of Spyder.

debugging, data exploration, interactive execution, deep inspection, and visualization. Besides software, the mechanism of method 1, the kernel-nearest neighbours (k-NN) algorithm, should be introduced before conducting the optimization process.

First coined by Cover and Hart [16], the k-NN algorithm is a non-parametric method to solve classification problems. It chooses several nearest samples as references to classify unknown ones. Firstly, distances between all known samples and the testing sample are calculated. Next, k (the number of) closest reference samples are chosen and checked which class they belong to, and then based on the majority rule, unknown samples are put in the specific sort. Though its result is sensitive to the k value set by people, the k-NN is a simple algorithm easy to be realised. More detailed information about k-NN can be found in other specific studies.

3.2. Optimization and results

What has been discussed above is the supervised classifying function of k-NN, but it is used here only for finding the nearest point. Initially, all 399 solutions are plotted as dots in the space composed by the total U value, the total heat capacity, and the total cost (Fig. 3). To be on the same scale with the other two extremely smaller parameters, material costs are reduced one thousand times. This action would not influence the solution finding because every solution is scaled at the same time with the same ratio.

Afterwards, authors could find the reference point (green point in Fig. 3), which has the minimum U value and the total price, but the maximum value of the heat capacity. It does not exist among the 399 points but an assumed point with its values from different solutions. The nearest point to this referencing one is the ideal solution searched for, which satisfies predefined objectives. In the present study, the defaulted uniformity on three axes means those three objectives have the same weighting factor.

Meanwhile, all the data is transformed into a fast indexing structure with the Ball-tree algorithm, which uses hypersphere to bound the Euclidean space [17]. It can find the nearest point faster than its counterparts as Brute force or KD tree. Afterwards, all the obtained distances are sorted, and the desired solution is illustrated as a green dot in Fig. 3. The obtained optimal solution is a combination of materials A1-B19-C2 (Table 4). Its total U value is 0.26, while the heat capacity is 1.18, with a price totalling to 465.5 Chinese yuan. The whole code of this finding process is given in the appendix.

4. Implementation in Grasshopper with an evolutionary algorithm

4.1. The software environment and the SPEA2

Besides the first approach introduced above, authors also tried to implement the idea using another method, which is mainly based on the Grasshopper, a built-in plugin of the 3D modelling software Rhino for visual programming (Fig. 4). Developed by Vierlinger [18], the Grasshopper's extension Octopus is the optimizer herein. The Octopus is the most accessible and popular multi-objective optimizer for architects. It has been downloaded more than 20,000 times on the website food4Rhino, which offers Grasshopper plugins.

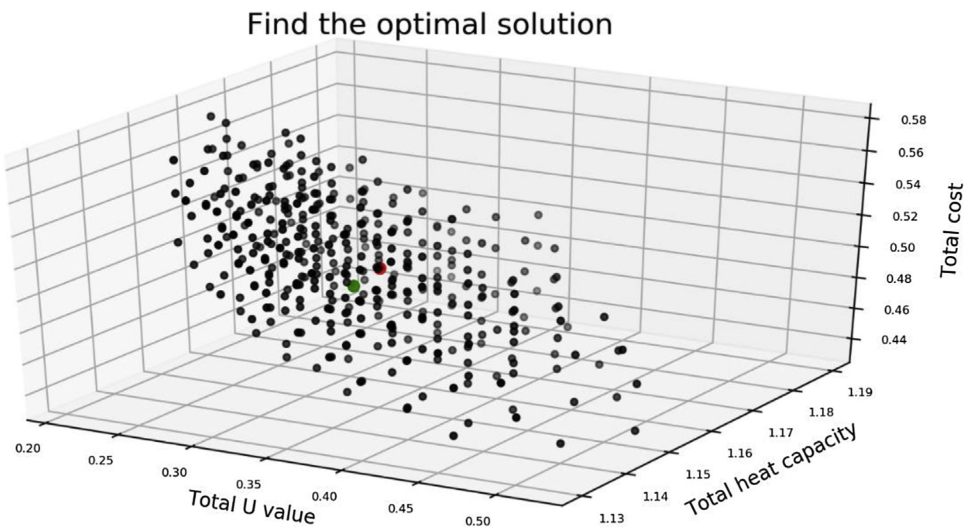

Fig. 3. The optimal result found with the k-NN.

Table 4
The optimal combination (A1-B19-C1) found by the K-NN.

Names	D (mm)	λ_n(W/m·K)	R_n(m^2·k/W)	C_{Pn}(kj/kg·K)	ρ_n(kg/m^3)	m_n(kg/m^2)	P_n(CN¥/m^2)
A1	40	0.53	0.08	1.05	75	3	350
B19	50	0.02	2.50	4.77	30	1.50	28
C2	175	0.16	1.09	1.13	520	91	87.50

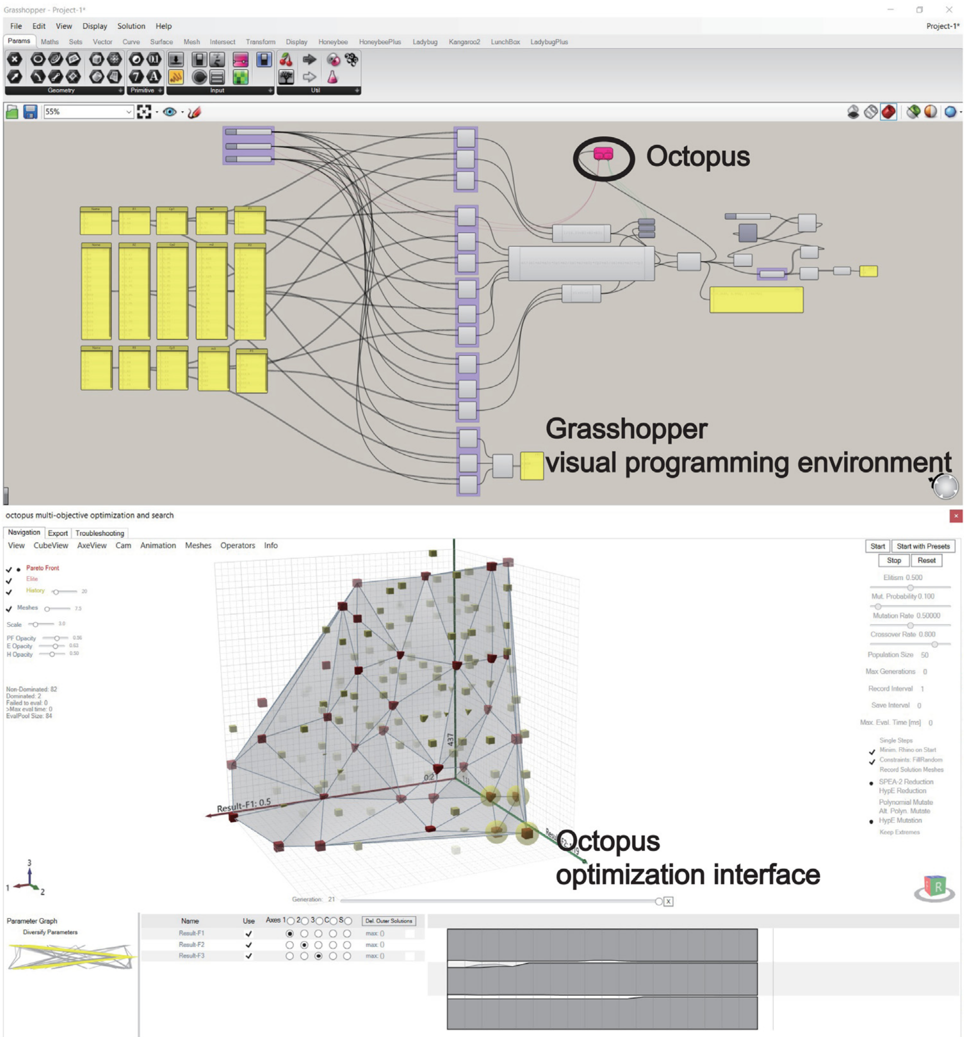

Fig. 4. Software environment of the method 2.

Unlike the first method directly taking the point closest to the pre-defined ideal solution is the best one among the alternatives, Octopus offers a Delaunay Pareto mesh, a surface composed of all Pareto solutions. Two terminology words need to be explained herein. A Delaunay triangulation for a set of discrete points in a plane is a triangulation such that no point from that set is inside the circumcircle of any triangle. The concept of Pareto-solution is that one solution is impossible to be relocated to make any one individual or preference criterion better off without making at least one individual or preference criterion worse off. We also can call the Pareto solution a non-dominated solution.

The optimization engine incorporated by the Octopus is the evolutionary algorithm SPEA2. SPEA is the acronym for Strength Pareto Evolutionary Algorithm, which was first proposed by Zitzler and Thiele [19] to find the Pareto-optimal set. An improved version namely SPEA2 was developed three years later, by integrating an improved fitness assignment scheme, a nearest neighbour density estimation technique for more precise search guidance, and a new archive truncation method. More detailed information of the SPEA2 could be found in that original publication written by Zitzler et al. [20].

4.2. Optimization model and results

Before starting the optimization process, a model needs to be built, correlating material properties with selection objectives. First of all, the material information is imported into Grasshopper as 15 panels (Fig. 5a, left). The first column is the materials' names, and the left four columns are information of four related variables R_n, C_{Pn}, m_n, and P_n. The first row is the information of ECP layer materials, while the other two rows are the insulation layer and the ALC board respectively. Afterwards, they are exported into the data structure type of list (Fig. 5a, right). Meanwhile, from these lists, the specific material information is indexed by three number sliders (Fig. 5b). Indexes from three sliders are encoded as genes in the Octopus (Fig. 5c, red lines), and one batch of them compose a chromosome for the evolutionary algorithm. Meanwhile, by calculating data from lists, three fitness functions produce three results. These three results are fed back to the Octopus (Fig. 5c, dark green lines) to guide the optimization process. At the same time, results from three fitness functions make up a point in the design solution space, and these spots are sent to the Octopus as the phenotype (solution appearance or properties). Designers can reinstate any solution from the Octopus interface and exhibit it (Fig. 5d). At last, a monitoring system is designed to record how many solutions have been calculated (Fig. 5e).

The population size in the current case study is set at 50, and the algorithm stopped at 21 generations, after investigating 315 solutions. The optimization result is shown in Fig. 6. Cubes in the opaque dark red stand for those non-dominated solutions, while ones in transparent red are dominated solutions but still belonging to the Elite. Transparent yellow cubes are elite solutions from the history, the more transparent, the older. As introduced in chapter 4.1, a Pareto mesh is generated with the dark grey colour. This mesh is composed of Pareto solutions (non-dominated ones). More balanced on simultaneously pursuing three objectives, four preferred solutions are selected by authors and marked with yellow spheres. Besides the combination A1-B19-C2 which was also found by method 1, the other three alternatives are A1-B19-C1, A1-B18-C1, and the A1-B18-C2. Due to the fact that these alternatives only have small differences on three objectives, designers can select any one of them based on personal preferences.

4.2.1. Usability test and discussions

To test the usability of these two methods, the current authors organized an online workshop with architects. All 15 participators had an architecture bachelor degree, and one-third of them had a master degree. Meanwhile, they all had 1–3 years working experience and also basic Grasshopper skills. The whole workshop was composed of three stages. During the first stage, architects were required to manually find an optimal material combination, with the same objectives in this case study. In the second phase, participators were asked to conduct the same mission but using the code provided by method 1 and method 2. Results from stage 1 and stage 2 were compared, concerning their compliance with three objectives and also the time consumption.

Confirmed by results from the workshop, method 2 is much more impressive than the manual approach and method 1. By using the manual approach, only two architects found a solution belonging to the preferred four solutions obtained in the

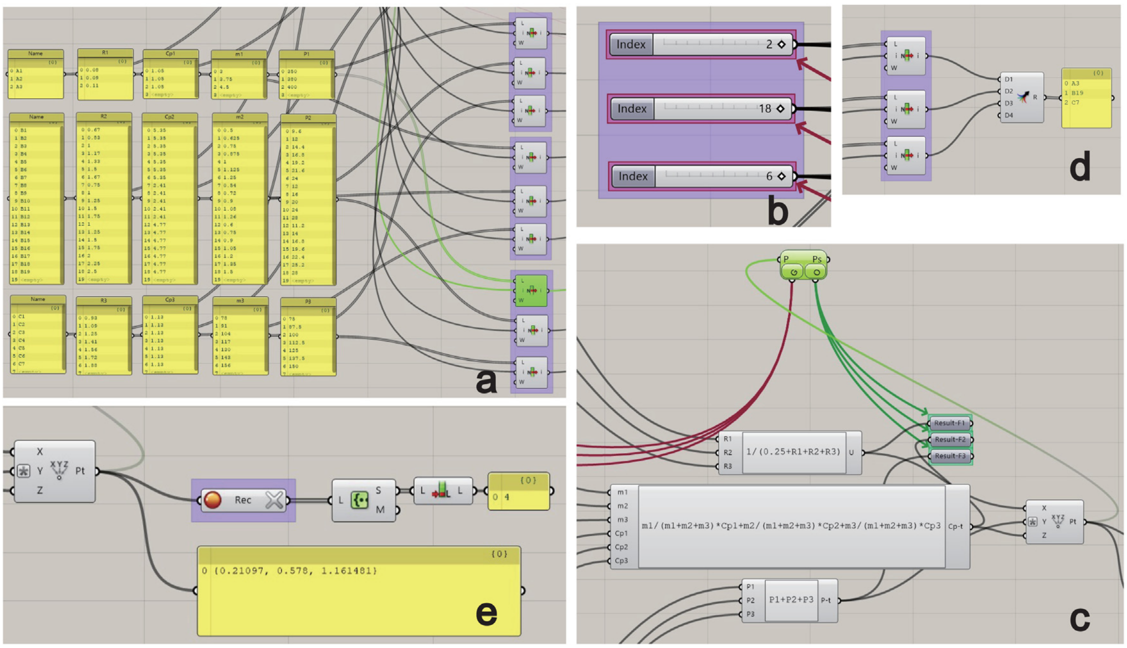

Fig. 5. The optimization model coded in Grasshopper.

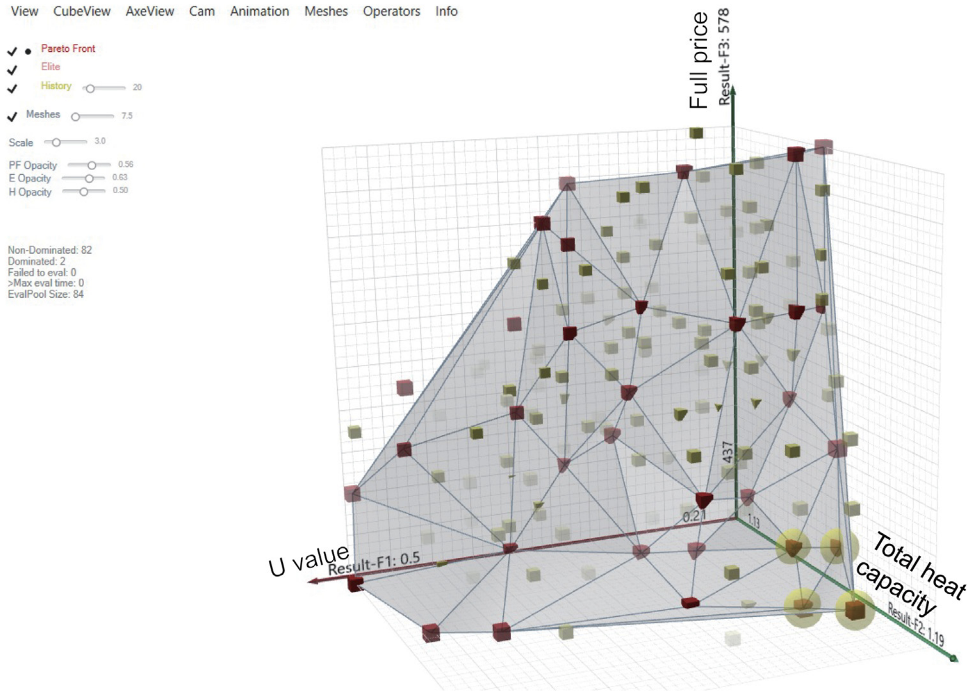

Fig. 6. Optimization results by the method 2.

previous chapter. Even with this result, the average time consumption of using the manual approach was half an hour more than later methods. In the last stage, participators were asked to fill out a questionnaire about their evaluation of this method. All people deemed that method 2 is architect-friendly and easy to operate. The main reason is that they are more familiar with the Grasshopper interface and the visual programming language. Meanwhile, the mathematic mechanism of the genetic algorithm, used in method 2, is easier to be roughly understood.

Although more efficient and architect-friendly than the manual approach, method 2 is still computationally inefficient compared to method 1, which only cost one second to complete the mission. If alternative material combinations increase, the computation time of method 1 will remain the same but the time of method 2 will rise. The reason is that Grasshopper's advancement is parametrically modelling the geometry, rather than purely managing the data. However, method 2 can integrate more related design issues. For example, we can draw the wall layer geometry in Rhino, assign materials in Grasshopper, also run the heat transfer analysis software THERM through the interface in Grasshopper, and then load simulation results directly in Rhino.

5. Limitations and future opportunities

There are several limitations of the current study, which also means research opportunities to enhance it. Firstly, the present case study takes the simplest case of a pre-fabricated wall, with only three material layers. In future, this method will be applied to walls with more layers, such as walls with a moisture barrier, etc. Moreover, not only walls, roofs and other building components can also be executed.

Meanwhile, more objectives will be integrated to find the optimal combination, for instance, time delay and decrement factor of the envelope [21,22], condensation risks, and also the mould-risk. It is possible to integrate these extra objectives. However, the visualization of high-dimension multi-objective optimization is more difficult. Meanwhile, the trade-off between incorporating more objectives and the increase of computation cost will be discussed first.

Speaking of the research method, only two algorithms are executed and compared herein. However, there are more algorithms can be utilized for this study, such as the Bubble Sort, Insertion Sort, etc. More research experiments will be conducted on this aspect.

6. Conclusions

Optimization technology is maturely utilized in other fields, but optimization studies on the architecture design are rarely applicable to different design projects. Meanwhile, the most precedent studies are not easy nor practical for common architects to use in design projects. Therefore, the current study utilizes the most available tools and simple algorithms to realize the material designation automatically. To obtain the optimal material combination, four parameters of material characteristics are correlated with three objectives. Authors experiment on this work with two different implementation methods: k-NN in Python and the SPEA2 in Grasshopper. These two methods were compared with the manual approach through a usability test workshop, and future opportunities are also highlighted.

The paradigm shift from manual selection to automatic finding reduces architects' repetitive work and helps them get better results. The current research has significance in filling the gap between the application of multi-objective optimization and architects' knowledge inadequacy.

Acknowledgements

This work is one part of author's PhD thesis, supported by the China Scholarship Council under the Grant 201607820005.

Appendix A. Supplementary data

Supplementary material related to this article can be found, in the online version, at doi:https://doi.org/10.1016/j.cscm.2019.e00220.

References

[1] R. Plag, Ubakus Retrieved January 9, 2019, from, (2019) . https://www.ubakus.de/u-wert-rechner/index.php?#.
[2] Ekotrope, R Value Calculator Retrieved January 9, 2019, from, (2019) . https://ekotrope.com/r-value-calculator/.
[3] Applied Building Technology Group, L, Wall Calculator Retrieved January 9, 2019, from, (2018) . https://www.appliedbuildingtech.com/fsc/calculator.
[4] R. Azari, S. Garshasbi, P. Amini, H. Rashed-Ali, Y. Mohammadi, Multi-objective optimization of building envelope design for life cycle environmental performance, Energy Build. 126 (2016) 524–534, doi:http://dx.doi.org/10.1016/j.enbuild.2016.05.054.
[5] D. Castro-Lacouture, J.A. Sefair, L. Flórez, A.L. Medaglia, Optimization model for the selection of materials using a LEED-based green building rating system in Colombia, Build. Environ. 44 (6) (2009) 1162–1170, doi:http://dx.doi.org/10.1016/j.buildenv.2008.08.009.
[6] E.K. Zavadskas, R. Bausys, B. Juodagalviene, I. Garnyte-Sapranaviciene, Model for residential house element and material selection by neutrosophic MULTIMOORA method, Eng. Appl. Artif. Intell. 64 (May) (2017) 315–324, doi:http://dx.doi.org/10.1016/j.engappai.2017.06.020.
[7] Wikipedia, Grasshopper 3D Retrieved January 17, 2019, from, (2018) . https://en.wikipedia.org/wiki/Grasshopper_3D.
[8] H.-S. Kim, S.-H. Lee, B. Kim, Properties of extrusion concrete panel using waste concrete powder, Appl. Sci. 7 (9) (2017) 910, doi:http://dx.doi.org/10.3390/app7090910.
[9] B.P. Jelle, Traditional, state-of-the-art and future thermal building insulation materials and solutions - properties, requirements and possibilities, Energy Build. 43 (10) (2011) 2549–2563, doi:http://dx.doi.org/10.1016/j.enbuild.2011.05.015.
[10] I.B. Topcu, T. Uygunoglu, Properties of autoclaved lightweight aggregate concrete, Build. Environ. 42 (12) (2007) 4108–4116, doi:http://dx.doi.org/10.1016/j.buildenv.2006.11.024.
[11] Y. Li, H. Wang, H. Qian, Y. Lei, Q. Bai, Effect of autoclaved lightweight concrete panels on the mechanical performance of container building and its equivalent method, Adv. Struct. Eng. 21 (7) (2018) 1103–1116, doi:http://dx.doi.org/10.1177/1369433217739706.
[12] P. Yang, K. Tang, X. Yao, Turning high-dimensional optimization into computationally expensive optimization, IEEE Trans. Evol. Comput. 22 (1) (2018) 143–156, doi:http://dx.doi.org/10.1109/TEVC.2017.2672689.
[13] S. Zhao, E. DeAngelis, Reduce mould risk during the building design stage: case studies in Southeast China (in press), J. Constr. Dev. Ctries. (2019) http://web.usm.my/jcdc/earlyView/JCDC%20Early%20View_Shenghuan%20Zhao.pdf.
[14] P.T. Tsilingiris, On the thermal time constant of structural walls, Appl. Therm. Eng. 24 (5–6) (2004) 743–757, doi:http://dx.doi.org/10.1016/j.applthermaleng.2003.10.015.
[15] C.W.Ip Kenneth, Critical thermal transmittance (U) value for the design of green buildings. ClB TG 16, Sustain. Conslmclion 36 (1994) 287–296, doi:http://dx.doi.org/10.1016/0140-6701(95)95641-H.
[16] T. Cover, P. Hart, Nearest neighbor pattern classification, IEEE Trans. Inf. Theory 13 (1) (1967) 21–27, doi:http://dx.doi.org/10.1109/TIT.1967.1053964.
[17] S.M. Omohundro, Five balltree construction algorithms, Science 51 (1) (1989) 1–22, doi:http://dx.doi.org/10.1016/S0092-8240(89)80047-3.
[18] R. Vierlinger, Multi Objective Design Interface 2013(April), (2013), pp. 1–61, doi:http://dx.doi.org/10.13140/RG.2.1.3401.0324.
[19] E. Zitzler, L. Thiele, An Evolutionary Algorithm for Multiobjective Optimization: The Strength Pareto Approach. TIK-Report https://doi.org/10.1.1.40.7696, (1998) .
[20] E. Zitzler, M. Laumanns, L. Thiele, SPEA2: Improving the Strength Pareto Evolutionary Algorithm Zurich, (2001), doi:http://dx.doi.org/10.3929/ETHZ-B-000225616.
[21] C. Sun, S. Shu, G. Ding, X. Zhang, X. Hu, Investigation of time lags and decrement factors for different building outside temperatures, Energy Build. 61 (2013) 1–7, doi:http://dx.doi.org/10.1016/j.enbuild.2013.02.003.
[22] D. Mazzeo, G. Oliveti, N. Arcuri, Influence of internal and external boundary conditions on the decrement factor and time lag heat flux of building walls in steady periodic regime, Appl. Energy 164 (2016) 509–531, doi:http://dx.doi.org/10.1016/j.apenergy.2015.11.076.