Artificial Intelligence for Engineering Design, Analysis and Manufacturing

cambridge.org/aie

Optimization test of a rule-based swarm intelligence simulation for the conceptual design process

Asli Agirbas

Fatih Sultan Mehmet Vakif University, Sutluce Mah. Karaagac Cad. No: 12/A Beyoglu, Istanbul, Turkey

Research Article

Cite this article: Agirbas A (2020). Optimization test of a rule-based swarm intelligence simulation for the conceptual design process. *Artificial Intelligence for Engineering Design, Analysis and Manufacturing* **34**, 477–491. https://doi.org/10.1017/S0890060420000323

Received: 29 December 2019
Revised: 22 June 2020
Accepted: 28 June 2020
First published online: 15 July 2020

Key words:
Bio-inspired design; conceptual design model; genetic algorithm; optimization; swarm intelligence

Author for correspondence:
Asli Agirbas, E-mail: asliagirbas@gmail.com

Abstract

Today, in the field of architecture, bio-inspired algorithms can be used to design and seek solutions to design problems. Two of the most popular algorithms are the genetic algorithm (GA) and swarm intelligence algorithm. However, no study has examined the simultaneous use of these two bio-inspired algorithms in the field of architecture. Therefore, this study aims to test whether these two bio-inspired algorithms can work together. To this end, GA is used in this study to optimize the rule-based swarm algorithm for the conceptual design process. In this optimization test, the objective was to increase the surface area, and the constraints are parcel boundary and building height. Further, the forms associated with swarm agents were determined as variables. Following the case studies, the study concludes that the two bio-inspired algorithms can effectively work together.

Introduction

Many scholars, such as Benyus (1998), Chakrabarti *et al.* (2005), Vincent *et al.* (2006), Chakrabarti and Shu (2010), Sartori *et al.* (2010), Vattam *et al.* (2010), Pawlyn (2011), and Pedersen Zari (2015), have interpreted bio-inspired thinking and production using different perspectives. Further, many researchers have long debated the immense potential of bio-inspired (or biomimetic) design. In addition, in the field of architecture, many buildings have been constructed based on the bio-inspired point of view. Today, architecture, particularly bio-inspired design, emphasizes design that rather than merely imitating, is based on the functions of the natural environment.

Natural functions can be mathematically analyzed and expressed using computer algorithms. Some of the popular bio-inspired algorithms are swarm intelligence algorithms and genetic algorithms (GAs), and these algorithms are used in the field of architecture to solve various problems. For example, Rajan (1995), Chatzi and Koumousis (2009), van Embden Andres *et al.* (2011), and Hofmeyer and Davila Delgado (2015) used GAs for structural optimization. Further, the GA was used by Jo and Gero (2006), Rawat *et al.* (2012), and Su and Yan (2015) in space planning; Wang *et al.* (2005), Attia *et al.* (2012), and Agirbas (2018) in green building design; and Mukerjee *et al.* (1997) in shape studies. Fourie and Groenwold (2002) and Felkner *et al.* (2013) are among the researchers who used particle swarm optimization (PSO), a slightly more advanced version of the swarm intelligence algorithm, in the field of structural optimization. Moreover, in their study, Hu *et al.* (2014) used the PSO to select the most appropriate machining scheme. In addition, hybrid algorithms based on the evolutionary algorithm have been extensively studied, for example, the study by Ahlquist *et al.* (2015).

There are general studies on the optimization of Reynolds' (1987) boids algorithm (rule-based swarm algorithm) with GAs, for example by Chen *et al.* (2006) and Alaliyat *et al.* (2014). There are also general studies on the optimization of Reynolds' (1987) boids algorithm with the PSO, for example, by Cui and Shi (2009) and Alaliyat *et al.* (2014). However, to date, no study has examined the simultaneous use of the swarm intelligence algorithm and GA in the field of architecture. However, the simultaneous use of these two algorithms can undoubtedly provide different problem solutions in the architectural world.

In the field of architecture, while there has been no review of the simultaneous use of the swarm algorithm and GA, there have been extensive reviews on the algorithms used in the field of building design (especially on optimization). For example, Attia *et al.* (2013), Evins (2013), Machairas *et al.* (2014), Nguyen *et al.* (2014), and Shi *et al.* (2016) reviewed the optimization methods in the field of building design. In addition, Wetter and Wright (2004), Elbeltagi *et al.* (2005), Wright and Alajmi (2005), Tuhus-Dubrow and Krarti (2009), Kämpf *et al.* (2010), and Hamdy *et al.* (2012) examined the performance of various algorithms in determining design solution.

The agents in the PSO algorithm move through the problem space in order to find the best solution, but Reynolds' (1987) boids algorithm (a rule-based swarm algorithm) does not perform optimization like the PSO algorithm. However, it is used for navigation and obstacle

© The Author(s), 2020. Published by Cambridge University Press

CAMBRIDGE UNIVERSITY PRESS

avoidance problems. In the conceptual design stage of architectural design, the features of this algorithm may conceivably be important for the production of form alternatives within the boundaries of the parcel, especially on the building scale. Therefore, this study focuses on this algorithm and uses the GA to list the results that can be achieved with this algorithm, in line with the objectives.

The main research question addressed by this study is as follows: Can the two bio-inspired algorithms work together in the field of architecture? Therefore, the hypothesis of this study is that rule-based swarm intelligence, which serves as a simulation, can be optimized with the GA. To test this hypothesis, the study optimized the rule-based swarm algorithm with the GA. This optimization test marks the first test of this hybrid algorithm. Therefore, the optimization was intentionally based on a simple architectural problem of increasing the surface area. Further, it was envisaged that architectural solutions would remain at the conceptual level.

This study addresses the following secondary research question: Can digital pathways be developed to provide more flexible solutions in the conceptual design phase? Today, the conceptual design stage involves the use of various computer programs. However, in terms of changing or incorporating new details, conceptual design, which is performed by these programs, is less flexible than manual sketching. Therefore, new methods to provide more flexible solutions should be developed. Accordingly, this study aims to develop a more efficient method for the conceptual design stage by using new technologies and computer programs.

Background

Conceptual design stage

According to Yu *et al.* (2015), architectural design is a non-linear process that enables the constant refinement of ideas. Goel (1992) lists the following phases of the architectural design process: problem structuring, preliminary design, refinement, and detailing. Preliminary design, which is a creative process, involves the consideration of several *ill-structured* design options until the desired shape is achieved. This is a conceptual design phase because, in this phase, the designer expresses their design using a sketch or an abstract and refrains from providing a detailed description.

In the conceptual design stage (particularly, the sketching stage at the very beginning of the design process), the designer considers many design possibilities (Schon, 1983; Garner, 1992; Schon and Wiggins, 1992; Goel, 1995; Suwa and Tversky, 1996; Lipson and Shpitalni, 2000; Do, 2002). In this process, the designer performs observations and makes visual documentation. The ideas developed during this creative process have a significant impact on the final product (Jin and Benami, 2010).

Constraints and objectives limit the number of solutions at the beginning of the design process; hence, it is better to define these aspects at the conceptual design stage. While producing form, designers consider both constraints and objectives, which are the facilitators of the design process. Otherwise, the number of design alternatives will increase, and designers will face difficulty in making decisions (in the absence of certain constraints, objectives, or rules). The following are feasible examples of design constraints: constraints imposed by material use, economic competence, daylight use, energy minimization, structural constraints, and the desired surface area, and limitations caused by the surrounding structures, parcel boundaries, floor height, and aesthetic concerns. The designers image their design according to these constraints and objectives and perform a type of optimization. However, the computational design offers facilities to make the optimization of the work easier for the designer.

Swarm intelligence

To clarify the complex swarm intelligence behavior, researchers often consider the natural behavior of social insects (Bonabeau *et al.*, 1999; Dorigo *et al.*, 2000; Camazine *et al.*, 2001; Theraulaz *et al.*, 2003; Dorigo and Stutzle, 2004). As examples of swarm behavior, researchers have examined the internal organization of ant colonies (Theraulaz and Bonabeau, 1995), bee colonies (Camazine, 1991), schools of fish (Partridge, 1982), and flocks of birds (Reynolds, 1987). These examples clarify that swarm behavior, which seems complex, is based on simple rules, and subsequently, these swarm intelligence rules can be converted into computer algorithms. In multi-agent system-based swarm intelligence algorithms, agents perform self-organization by exhibiting autonomous behavior. This study uses a flocking model that is based on the model referred to as a "boid" by Reynolds (2017). According to Reynolds (2017), the complex behavior of birds in their natural environment depends on three mathematical rules: alignment, separation, and cohesion. When birds in a flock fly in the same direction, each bird maintains a certain distance from its neighboring birds, and this is called the separation rule. Further, each bird follows a direction that is determined by the average overall orientation of the flock, and this is the alignment rule. Finally, each bird takes its position relative to the average position, which is the cohesion rule. Therefore, although each bird acts in accordance with its neighbor's environment, the final result is that each bird affects the swarm's geometry and self-organization. Therefore, long-term swarm behaviur and movement remain unpredictable (Fig. 1).

The swarm algorithm has been further developed and used in optimization studies. The PSO is the swarm algorithm that provides optimization (Eberhart and Kennedy, 1995; Kennedy and Eberhart, 1995; Eberhart *et al.*, 1996). In addition, many studies have examined this algorithm in detail (Reddy and Kumar, 2007; Kramar *et al.*, 2015; Adeniran and El Ferik, 2017; Ab Rashid *et al.*, 2019), and in general, most of these studies are results oriented.

Genetic algorithms

Today, in the field of computational design, optimization is generally performed using GAs (Goldberg and Holland, 1988; Frazer, 1995; Bentley and Wakefield, 1997; DeLanda, 2002). To date, many studies have examined GAs in detail (Rasheed and Hirsh, 1999; Hofmeyer and Davila Delgado, 2015; Su and Yan, 2015).

Using the principles of evolutionary biology, inheritance, mutation, selection, and crossover, GAs offer solutions to various design problems in a short time (Renner and Ekárt, 2003; Kim, 2013; Yu *et al.*, 2014). These algorithms, which have very complicated programming structures, can easily be used through the Galapagos program developed by David Rutten (Rutten, 2013; González and Fiorito, 2015).

Methods

Constraints, variables, and objectives

For the developed model, the parcel boundary and building height boundary were defined as the constraints. Further, the avoidance

https://doi.org/10.1017/S0890060420000323 Published online by Cambridge University Press

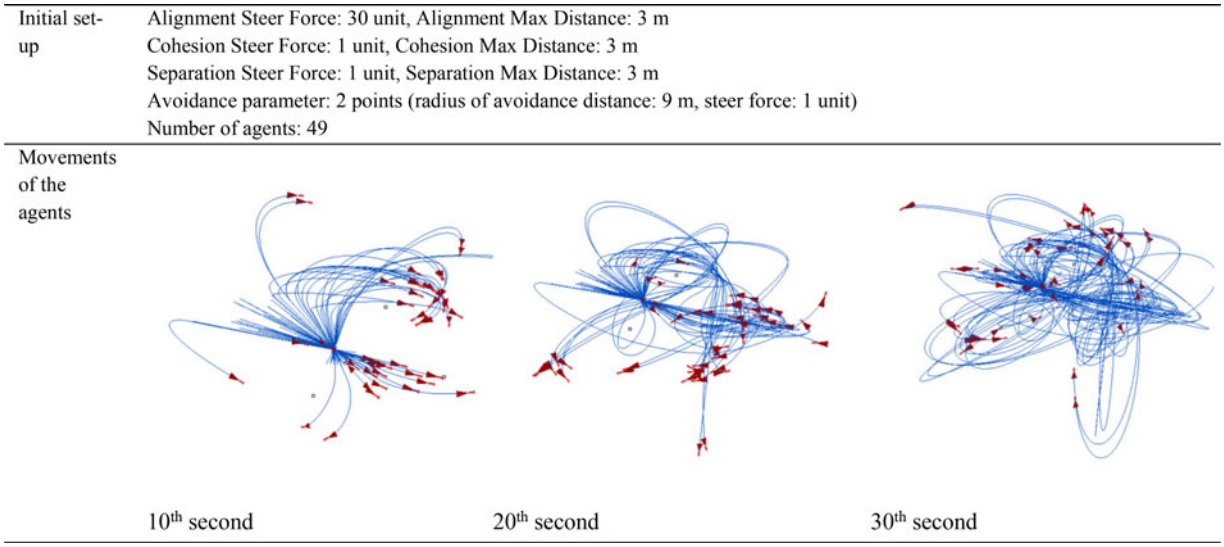


Fig. 1. Self-organization of agents according to the defined alignment, separation, and cohesion values.

points connected to the swarm simulation were the constraints, as they are related to the boundary limits of the site. In addition, the number and location of the agents, which were also related to the boundary limits of the site and whose values were maintained constant throughout the simulations, could also be defined as constraints. The objective was the enhancement of the surface area of the Delaunay or Voronoi-based form (Figs. 2, 3).

The variables were the values related to alignment, cohesion, and separation that determined the paths of agents in the swarm simulations. However, as the swarm geometry is directly determined by the agents, we observed the geometry associated with the form in the simulation. Therefore, the Delaunay or Voronoi-based form, which is connected to the swarm agents, was considered a variable (Fig. 2). The Delaunay mesh form (Lee and Schachter, 1980) and Voronoi-based form (Benvenuti *et al.*, 2013; Tedeschi, 2014) were separately connected to the agents in the swarm intelligence simulation. Delaunay and Voronoi geometries were used because they can provide form as a product in a certain complexity; they are directly connected to swarm agents (and, hence, to point geometry), that is, they do not complicate the simulation process; and form certain surfaces (or geometries that can become surfaces). However, other geometries with the same characteristics can be used in the simulations as well.

Computer software

In this study, all of the processes were performed in Grasshopper, which works as a plug-in to the Rhino program and provides a visual programming language environment. In addition, the study used Locust, Weaverbird, and Galapagos as add-ons to Grasshopper.

Model setup

For the swarm intelligence simulation, the study used Locust, which was developed as an add-on for the Grasshopper program created by Cerver Design studio (Cerver tools, 2018). First, a "bounding box" component was connected to the Locust Brain component in the Locust add-on. This box component was calibrated according to parcel boundaries and building height, and the simulation boundaries were defined. The height of the box was set as 12 m, as the assumed maximum achievable building height was 12 m. Further, the box's width was set as 12 m and the length as 16 m, which were in line with the boundaries of the parcel's usage area. In this manner, the movement boundaries of the agents in the swarm simulation were determined (Figs. 2, 4). Later, the starting points of the agents were connected to the Locust Brain component. These points were positioned on the ground plane and distributed in an equal ratio to the parcel. The designer determined the number of starting points, which is also the number of agents, and the distribution of these points according to the parcel's size. A grid was created by making five partitions each in the X and Y axes, and the points on the grid were defined as starting points (25 agents). In addition, while determining the partition ratio, it was assumed that the simulation could be run easily, and there were sufficient agents to create the desired form. The aforementioned number of partitions can be changed according to the formal complexity that can be achieved based on the number of agents and the parcel size.

The alignment, cohesion, and separation parameters, which correspond with the three main rules of swarm intelligence, can be separately inputted to the Locust Brain component. Further, there are two input values for alignment, cohesion, and separation; steer force and distance. Steer force specifies how power influences alignment, cohesion, or separation; and distance indicates the maximum distance between neighboring agents for alignment, cohesion, or separation. In this study, the steer force value is between 1 and 3 units (as a floating point), and the distance value is between 1 and 15 m (as a floating point) for alignment; the steer force value is between 1 and 3 units (as a floating point), and the distance value is between 1 and 15 m (as a floating point) for cohesion; and the steer force value is between 1 and 3 units (as a floating point), and the distance value is between 1 and 15 m (as a floating point) for separation (Table 1). However, these values can be assigned randomly or lie within very broad ranges. This is because Galapagos, which was used later in the script, finds the best value range. The values of these inputs can be changed parametrically, as well. The changes in these values cause variations in the overall behaviors of the swarm agents, as the former

https://doi.org/10.1017/S0890060420000323 Published online by Cambridge University Press

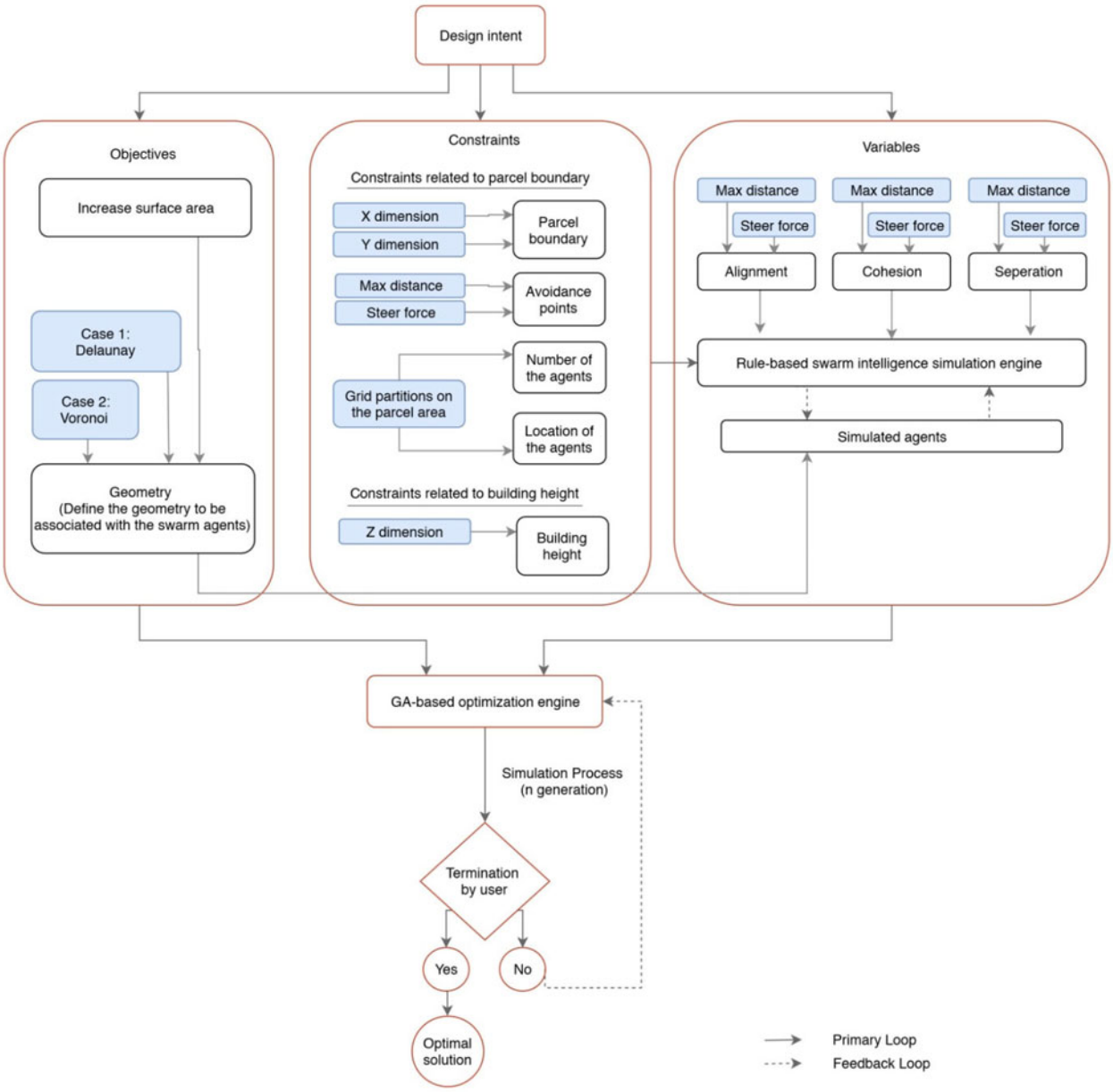


Fig. 2. Flowchart of the simulation script to maximize the surface area of the Delaunay/Voronoi mesh created by swarm agents.

may cause variations in the minimum and maximum distances among the agents. However, the probability combination values of these input values can be very high. In particular, the combination probability values increase with an increase in the number value interval (Fig. 2).

In addition, the avoidance point parameter can be added to the Locust Brain component. This parameter allows the agents to escape from specified points. Within the scope of this study, two avoidance points were defined in the considered area. These points were located in the middle of the buildings next to the parcel. The avoidance parameter has two inputs, steer force and distance. Based on the influence of the power of the agents moving away from the specified points, the steer force value was assigned the default value of 1 unit. Further, based on the impact of the distance of the agents moving away from the specified points, the distance parameter was set as 9 m. Therefore, the agents in the swarm simulation moved away from these points (i.e., from the buildings on the side). Based on the desired design, the script may contain different numbers of avoidance points in different locations (Fig. 2).

The agents can be identified as points in the swarm intelligence simulation output. A Delaunay mesh component was initially used, which is a Grasshopper component because it enables the creation of a complex form with ease by providing the points as inputs and involves a mesh geometry that allows the simulation to be performed at high speeds. The point geometry, that is, the input of the Delaunay mesh component, was connected to the swarm agents, which were the output of the swarm intelligence simulation. Hence, based on the simulation of the swarm agents, Delaunay mesh geometries were created. For the second experiment, the study used Voronoi geometry, which is, again, complex geometry. The Voronoi geometry was divided using the Catmull–Clark subdivision component, which is a component of the Weaverbird add-on. Subsequently, a thickness value

https://doi.org/10.1017/S0890060420000323 Published online by Cambridge University Press

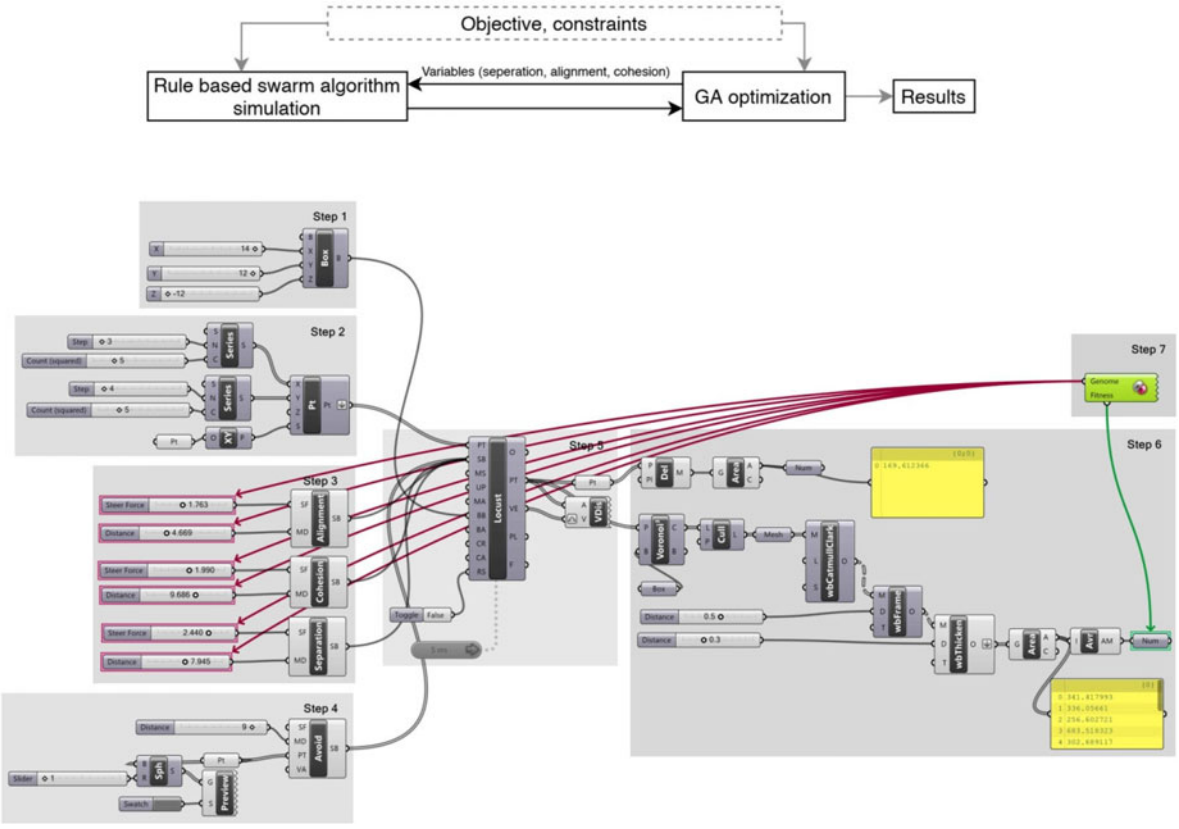


Fig. 3. The developed script.

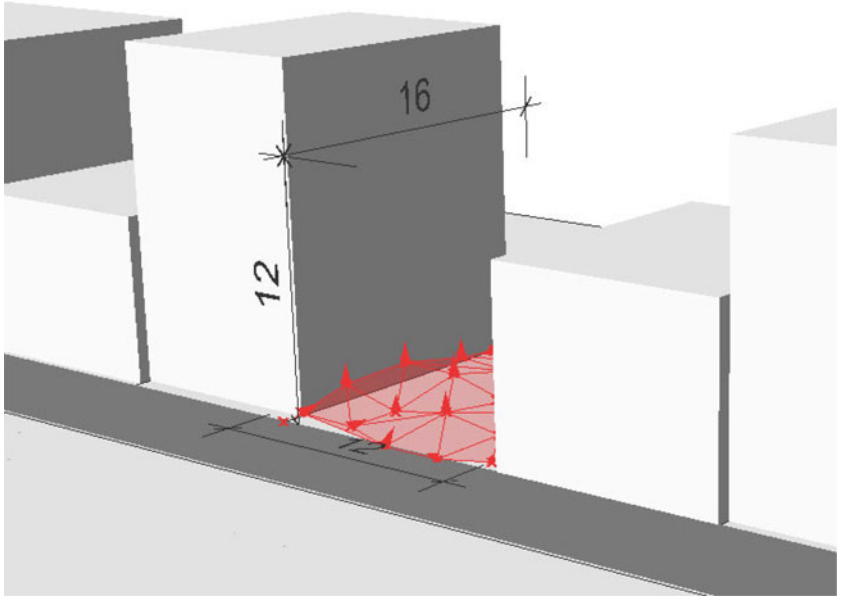


Fig. 4. The parcel and specified constraints (the arrows highlighted in red represent swarm agents; and the building height limit is 12 m, parcel boundaries are 12 m to 16 m).

was input to the division frame to provide a more complex mesh. The point input of the Voronoi component, which is the same as the one used in the Delaunay geometry, was connected to the agents in the point geometry, which was the output of the swarm intelligence simulation. In this manner, according to the simulation of the swarm agents, different Voronoi mesh geometries were created (Fig. 2).

Optimization simulation

In this study, for a particular parcel, the maximization of the surface area of the Delaunay/Voronoi mesh created by the agents throughout the simulation was defined as the criterion. This enabled the testing of how the mesh can be placed in the parcel with the maximum surface area at the conceptual level.

https://doi.org/10.1017/S0890060420000323 Published online by Cambridge University Press

Table 1. Results depicting the maximization of the area of the Delaunay mesh within the specified constraints

			Alignment		Cohesion		Separation	
Experiment no.	Fitness value (maximum area, m^2)	Number of generations created	Steer force (Range:1–3 units)	Max distance (Range: 1–15 m)	Steer force (Range:1–3 units)	Max distance (Range: 1–15 m)	Steer force (Range: 1–3 units)	Max distance (Range: 1–15 m)
1	413.64	51	1.76	13.23	1.7	5.13	2.24	11.58
2	399.59	49	2.8	11.06	1.67	12.8	1.53	5.61
3	505.61	91	2.34	6.62	2.03	7.9	2.57	9.66
4	253.16	49	2.1	9.21	2.71	8.05	1.78	9.07
5	404.93	52	1.96	7.47	2.14	2.67	1.73	6.14
6	387.83	49	1.53	8.2	1.96	14.58	1.25	6.37
7	381.45	49	1.32	10.84	1.41	2.5	1.42	12.23
8	331.57	57	1.72	8.06	2.63	5.38	1.69	7.1
9	360.31	49	2.74	5.67	2.57	5.51	1.05	5.73
10	345.04	49	2.37	12.31	2.52	13.27	1.9	1.65
11	367.65	49	2.67	2.89	2.25	8.13	1.1	8.51
12	526.89	124	1.73	8.15	1.83	8.23	2.73	7.22
13	523.20	62	1.72	8.25	1.94	8.5	2.14	7.02
14	409.99	49	2.22	8.77	2.99	9.31	1.14	8.06
15	448.46	146	2.33	10.72	1.71	3.35	1.86	7.21
16	529.01	73	2.16	6.35	1.74	5.83	2.35	6.55
17	452.64	54	2.32	9.41	1.55	4.98	2.48	8.79
18	498.72	123	2.32	4.05	1.65	5.32	1.68	11.87
19	548.54	50	2.35	4.22	1.96	6.98	2.54	8.02
20	401.89	49	2.74	1.88	1.52	3.46	1.26	3.93

However, it is very difficult for this formation to capture the combination of the agent parameters that form the maximum surface area by adjusting the number of slider parameters (the values of inputs of the alignment, cohesion, and separation components) and then running the simulation. Therefore, Galapagos, which was developed as an add-on for Grasshopper based on GA, was used as an optimization tool to determine the options for the most suitable values. The genome part of Galapagos (representing the variables to be used to reach the fitness value) was connected to the number value parameters (six inputs) of the steer force and distance inputs of alignment, cohesion, and separation. Further, the numerical value of the Delaunay mesh surface/Voronoi mesh average surface area to be formed as a result of the swarm simulation was connected to the fitness part. Subsequently, the Galapagos optimization was initiated without a runtime limit, and the process was set to maximize the value (initial settings for the optimization process: maximum stagnant: 50, population: 50, initial boost: 2, maintain: 5%, and inbreeding: +75%; Fig. 2).

Results and discussion

Case 1: Delaunay mesh

Using the Galapagos program, 20 experiments were conducted to identify the combination of alignment, cohesion, and separation values that increases the surface area of the Delaunay mesh (Table 1). In the first experiment, the alignment, cohesion, and separation values were randomly assigned before starting the simulation. In the experiments, the combination of alignment, cohesion, and separation values identified by the Galapagos program at the end of each experiment was used as the combination of the initial values for these parameters in the next experiment. This is because, in the swarm simulation, the combination of the initial values of the agents is important, as it allows the agents to influence each other's organization throughout the simulation process. In addition, more than one generation was produced in each of these 20 experiments to achieve the optimized result. At least 49 generations were produced in each of these experiments, and the number of generations produced in each experiment is illustrated in Table 1.

At the conclusion of all of the experiments, the combinations of the alignment, cohesion, and separation values determined by the Galapagos program to maximize the surface area value were found to be different from each other. An important reason is that the organization of swarm agents among themselves in each combination was different (in other words, each simulation recorded different movements of the agents), providing different Delaunay geometries for different combinations as well. However, all of the experiments provided area values between 253.16 and 548.54 m^2 (Table 1). Hence, in the model, each of the 20 experiments provided information about the maximum surface area of the corresponding form, which could be created with different combinations of the swarm simulation. This enabled the designer to identify the combination accurately of

https://doi.org/10.1017/S0890060420000323 Published online by Cambridge University Press

the starting values at which the maximum surface area could be reached.

A detailed examination of these 20 experiments reveals the following: The steer force values range between 1 and 3 units; alignment steer force has a maximum value of 2.80 unit and a minimum of 1.32 unit; cohesion steer force value ranges between the upper and lower limits of 2.99 unit and 1.41 unit, respectively; and separation steer force value varies from 2.73 unit to 1.05 unit. In other words, the values can vary significantly according to different combinations. Further, based on the limited distance values between 1 and 15 m, the alignment distance has a maximum value of 13.23 m and a minimum of 1.88 m, the range of the cohesion distance value is 14.58–2.50 m, and the range of the separation distance value is 12.23–1.65 m. In other words, the distance values can vary extensively according to different combinations. However, the value of one parameter does not increase in response to an increase in the values of other parameters. On the contrary, the other parameters have smaller values. For example, in experiment 1, the separation, alignment, and cohesion distances were 11.58 m, 13.23 m, and 5.13 m, respectively. Similarly, in experiment 2, the cohesion, alignment, and separation distances were 12.80 m, 11.06 m, and 5.61 m, respectively (Table 1).

In terms of maximizing the surface area, the aforementioned values, which were obtained from the model as a result of optimization, can be utilized to achieve a more efficient form-finding process. Accordingly, an experiment in which a swarm simulation was initiated with the values obtained at the end of the initial optimization was performed. As seen from the simulation process, the values of the Delaunay mesh surface areas tended to increase (Table 2), and the fluctuations in the area value were caused by the movements of agents in the simulation and, hence, considered normal.

Case 2: Voronoi mesh

Using the Galapagos program, 20 experiments were conducted to identify the combination of alignment, cohesion, and separation values that increases the average surface area of the Voronoi mesh. A process similar to the one performed in the Delaunay mesh experiment was followed. Further, similar to the Delaunay mesh case, different combinations of the alignment, cohesion, and separation values were determined by the Galapagos program to achieve the maximum surface area value. The major reason is that the organization of swarm agents among themselves in each combination was different, such that each combination formed a different Voronoi geometry. However, the experiments provided area values between 420.15 and 1261.74 m^2 (Table 3), and at least 25 generations were produced in each of these experiments. The number of generations produced in each experiment is illustrated in Table 3.

A detailed examination of the results of the 20 experiments reveals the following: the steer force values range between 1 and 3 units, the alignment steer force has a maximum value of 2.70 unit and a minimum of 1.00 unit, the cohesion steer force value is between 2.58 and 1.66 units, and the separation steer force value is between 2.73 and 1.45 units. In other words, the values vary significantly according to the different combinations. From the limited distance values between 1 and 15 m, the alignment distance value varies between 13.55 and 1.93 m, the cohesion distance value is between 14.14 and 4.14 m, and the separation distance value is between 11.32 and 3.06 m. In other words, the distance values vary significantly according to the different combinations. However, similar to the Delaunay mesh experiment, the value of one parameter does not increase in response to an increase in the values of other parameters. In contrast, the other parameters have smaller values. For example, in experiment 1, the separation, alignment, and cohesion distances are 7.94 m, 4.66 m, and 9.68 m, respectively. Similarly, in experiment 2, the cohesion, alignment, and separation distances are 8.97 m, 10.67 m, and 11.32 m, respectively (Table 3).

Similar to the case of the Delaunay mesh, an experiment in which a swarm simulation was initiated with the values obtained at the end of the initial optimization was conducted for the Voronoi mesh. As seen from the simulation process, the values of Voronoi mesh surface areas tend to increase with the progress of the experiment (Table 4). The fluctuations in the area value were caused by the movements of agents in the simulation and, hence, considered normal.

The number of agents in the swarm simulation; locations of the agents; and initial values of the alignment, cohesion, and separation parameters are the factors determining how a swarm simulation will formally occur. In the experiments conducted in this study, the number and locations of agents were kept constant. The simulation was initiated with random alignment, cohesion, and separation values. At the end of the experiments, GA optimization was performed to identify the combination of alignment, cohesion, and separation values in the swarm simulation that created the maximum surface area values.

Opposite objective test

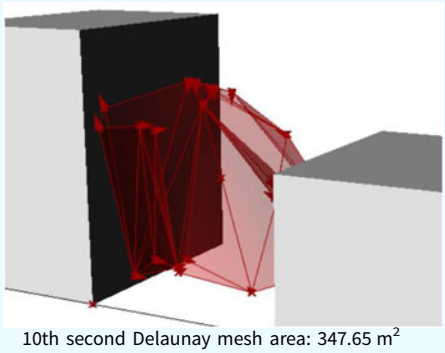
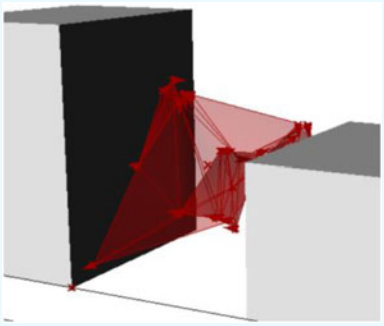
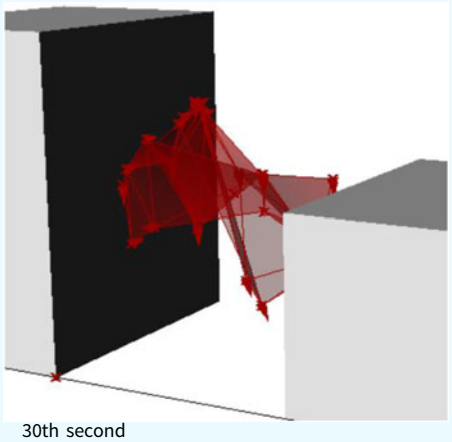
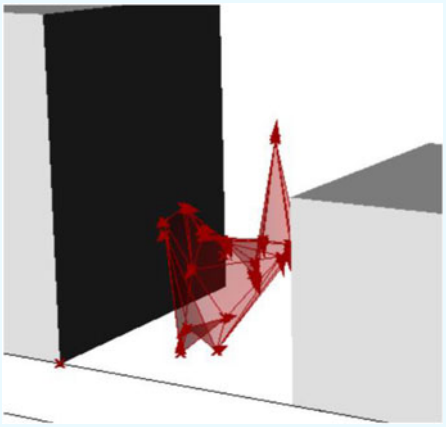
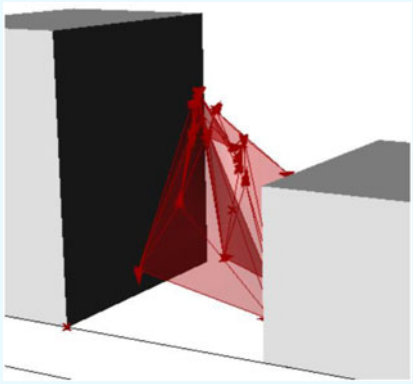
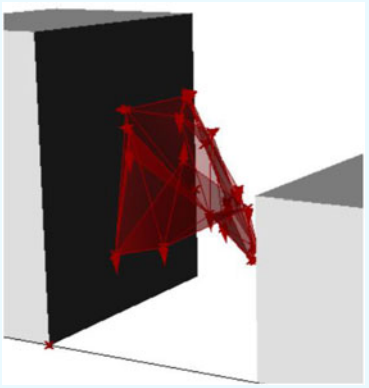
In this study, to test the accuracy of the information produced by the proposed model, a process to minimize the surface area of the Delaunay mesh produced by swarm simulation was determined as a criterion, and a test operation was carried out. In this process, apart from applying the settings to minimize the area value of the objective (criterion) in the optimization step, the constraints and methodology were exactly the same as the process for determining the maximum surface area.

This process aimed to test the accuracy of the information included in the model by performing an opposite operation. In this process, test experiments for the Delaunay mesh were conducted to examine whether the forms, which could be formed in a swarm simulation and could cover the minimal area, occurred or not. The test trials revealed that forms with very low area values could be obtained by swarm simulation optimization (Table 5), and the model took only a short time to provide this information. All of the experiments revealed area values between 104.21 and 0 m^2. In the swarm simulation, when the agents came very close to each other and became concentrated in some regions, the area value can fall to 0, which is reflected in the results. Hence, the model could provide information on the minimum surface area of the form that resulted from the swarm simulation in different combinations in each of the 20 experiments. This assisted the designer in identifying the combination of starting values that could yield the minimum surface area.

A detailed examination of the 20 test results reveals that among the steer force values that are limited between 1 and 3 units, the values of the alignment steer force have a maximum of 2.63 unit, and a minimum of 1.49 unit, the cohesion steer force value has a maximum of 2.68 unit and a minimum of 1.23 unit, and the separation steer force value is at a maximum 2.78 unit and a

https://doi.org/10.1017/S0890060420000323 Published online by Cambridge University Press

Table 2. Sections of the swarm simulation initiated by the values obtained from the first Galapagos optimization for the Delaunay mesh area

Initial setup	Alignment steer force: 1.76 unit, alignment max distance: 13.23 m
	Cohesion steer force: 1.7 unit, cohesion max distance: 5.13 m
	Separation steer force: 2.24 unit, separation max distance: 11.58 m
	Avoidance parameter: 2 points (radius of avoidance distance: 9 m, steer force: 1 unit)
	Number of agents: 25 (X:5, Y:5)
	Box: 12 × 16 × 12 m
Swarm simulation results	 10th second Delaunay mesh area: 347.65 m^2
	 20th second Delaunay mesh area: 218.12 m^2
	 30th second Delaunay mesh area: 184.3 m^2
	 40th second Delaunay mesh area: 88.99 m^2
	 50th second Delaunay mesh area: 227.17 m^2
	 60th second Delaunay mesh area: 146.36 m^2

minimum of 1.14 unit. In other words, the values can vary significantly among the combinations. From the limited distance values between 1 and 15 m, the alignment distance has a maximum value of 14.76 m and a minimum of 2.93 m, the cohesion distance has a maximum value of 12.03 m and a minimum of 1.17 m, and the separation distance has a maximum value of 12.15 m and a minimum of 3.04 m. Finally, once again, the distance values varied extensively for different combinations (Table 5).

https://doi.org/10.1017/S0890060420000323 Published online by Cambridge University Press

Table 3. Results of maximizing the area of the Voronoi mesh within the specified constraints

			Alignment		Cohesion		Separation	
Experiment no.	Fitness value (maximum area, m^2)	Number of generations created	Steer force (Range: 1–3 units)	Max distance (Range: 1–15 m)	Steer force (Range: 1–3 units)	Max distance (Range: 1–15 m)	Steer force (Range: 1–3 units)	Max distance (Range: 1–15 m)
1	512.94	63	1.76	4.66	1.99	9.68	2.44	7.94
2	838.17	59	2.26	10.67	2.12	8.97	2.09	11.32
3	780.99	67	2.15	10.93	2.12	8.53	1.75	10.6
4	732.38	89	2.02	10.15	2.04	8.93	1.8	9.6
5	1226.26	40	2	9.74	2.02	8.95	1.82	9.57
6	923.08	84	2.66	2.15	1.66	14.14	2.73	9.5
7	455.94	131	1.55	12.91	2.58	8.24	2.7	8.98
8	420.15	90	2.7	8.39	2.25	10.24	2.61	7.08
9	424.34	44	1.1	1.93	2.17	11.98	1.64	6.82
10	823.21	143	2.31	13.55	2.03	4.14	1.45	11.06
11	434.42	58	2.15	5.84	1.95	7.53	1.94	9.3
12	437.97	64	2.66	4.48	1.86	10.5	2.33	9.01
13	453.84	75	1.62	9.29	2.36	5.38	2.2	8.74
14	461.11	25	1.47	11.3	2.02	9.99	1.69	3.06
15	422.29	60	2.37	7.03	1.88	7.07	1.69	4.18
16	887.66	56	2.33	5.82	2.44	6.31	2.1	9.93
17	974.03	95	2.3	7.23	2.32	5.93	2.05	9.01
18	947.85	44	1.67	8.1	2.28	6.32	2.08	8.29
19	1261.74	46	1.66	8.1	2.28	6.32	2.08	9.62
20	861.02	184	1	8.57	2.33	9.41	2.18	7.57

On concluding the experiments that were part of the test, the combination of alignment, cohesion, and separation values in the swarm simulation that created the minimum surface areas could be identified using GA optimization.

This test examined whether the script can work effectively by using opposite objectives (minimizing surface area). All of the other values in the main test were kept the same, and as a result, the script could perform the minimizing surface area task, which is the opposite objective. In particular, although the value ranges of the variables (the values related to alignment, cohesion, and separation) are the same with the main test, the decrease in fitness values obtained as a result of GA optimization demonstrated that the script could perform the opposite task. In addition, in the swarm simulation, which began by taking the values in the third experiment, the decrease of the area covered by the Delaunay form is noticeable in a short time (Table 6).

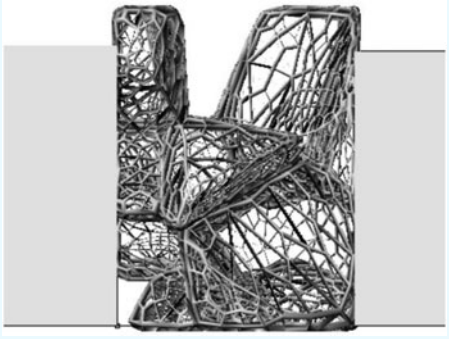
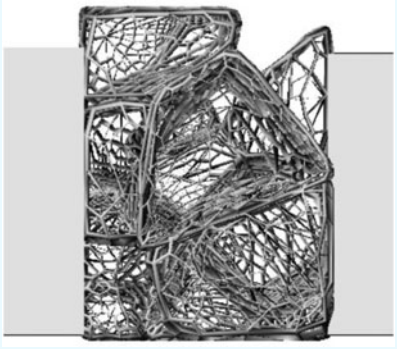
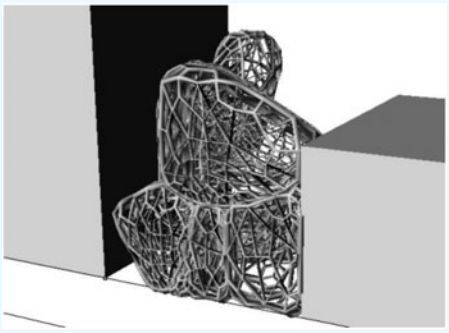
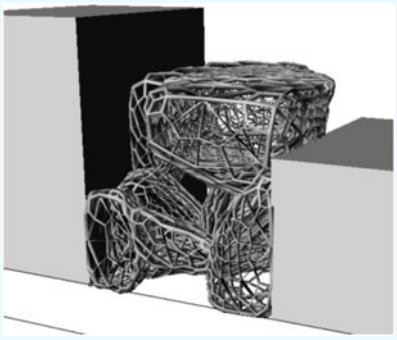
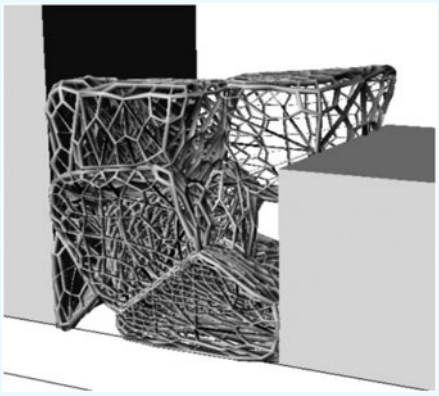
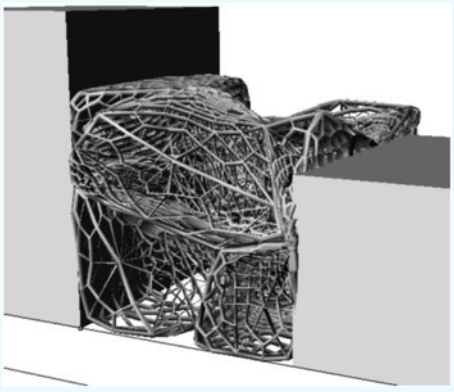
Analysis of the relationship between the values and the surface area

Another analysis was made to observe how the alignment, cohesion, and separation values affect the surface area of the form in the swarm simulation. To understand the effect in the Delaunay mesh form, first, a simulation experiment was carried out with the alignment steer force value: 1, alignment maximum distance value (m): 2, cohesion steer force value: 1, cohesion maximum distance value (m): 2, separation steer force value: 1, and separation maximum distance value (m): 2. Then, keeping the other values constant, an experiment with an increased alignment maximum distance value of 5 m, an experiment with an increased cohesion maximum distance value of 5 m, and another experiment with an increased separation maximum distance value of 5 m were conducted. This experiment was performed for the first 5 and 10 s because beyond this time, the form was found to be influenced by environmental factors owing to the effect of the bounding box. Therefore, the change could not be clearly observed (Table 7).

According to this analysis, the separation maximum distance value variable has the potential to increase the area of the form. If the environmental factors had no effect on shaping the form, it would be reasonable to retain the separation maximum distance value variable high to maximize the surface area because the separation feature enlarges the distance between the agents, and therefore increases the surface area formed between the agents. However, in this study, this is not a criterion as the environmental factors (sidewalls) affect the swarm simulation, and the swarm agents are self-organized in line with these environmental factors. However, GA optimization offers us a potential value combination (incorporation with environmental factors into the process), so we can obtain a form that may emerge during the swarm simulation. However, there may be increases and decreases in the surface area during the simulation period of this combination.

https://doi.org/10.1017/S0890060420000323 Published online by Cambridge University Press

Table 4. Sections of the swarm simulation initiated by the values achieved from the first Galapagos optimization for the average Voronoi mesh area

Initial setup	Alignment steer force: 1.76 unit, alignment max distance: 4.66 m	
	Cohesion steer force: 1.99 unit, cohesion max distance: 9.68 m	
	Separation steer force: 2.44 unit, separation max distance: 7.94 m	
	Avoidance parameter: 2 points (radius of avoidance distance: 9 m, steer force: 1 unit)	
	Number of agents: 25 (X:5, Y:5)	
	Box: 12 × 16 × 12 m	
Swarm simulation results	 Plan view	 Plan view
	 Perspective view 10th second Voronoi mesh average area: 253.54 m^2	 Perspective view 20th second Voronoi mesh average area: 261.67 m^2
	 30th second Voronoi mesh average area: 286.9 m^2	 40th second Voronoi mesh average area: 358.49 m^2

It should be noted that, in these simulations, the form changes every second. Therefore, depending on the computer used, the results may vary. The computer used in this study is a portable workstation (equipped with the Intel Core i7 – 3820QM processor – 8 MB Cache up to 3.70 GHz, 8 GB RAM, 500 GB Hard Disk Drive, 7200 rpm, and a separate NVIDIA Quadro K2000M graphics processor with 2GB DDR3 Memory). In the rule-based swarm intelligence simulation, 6% CPU was used, and in the combined simulation (swarm + GA optimization), 17% CPU was used.

https://doi.org/10.1017/S0890060420000323 Published online by Cambridge University Press

Table 5. Results of the minimization of the area of the Delaunay mesh (fitness value) within the specified constraints

			Alignment		Cohesion		Separation	
Experiment no.	Fitness value (minimum area, m^2)	Number of generations created	Steer force (Range: 1–3 units)	Max distance (Range: 1–15 m)	Steer force (Range: 1–3 units)	Max distance (Range: 1–15 m)	Steer force (Range: 1–3 units)	Max distance (Range: 1–15 m)
1	104.21	70	2.34	8.91	1.23	5.63	1.53	9.66
2	0	32	2.46	10.06	2.18	1.45	1.27	6.49
3	0.01	29	2.03	5.99	2.21	5.97	1.75	5.97
4	0	23	2.33	7.65	2.3	6.49	1.69	8.42
5	28.99	41	1.91	5.93	1.84	12.03	1.91	7.17
6	0.06	21	2.05	7.89	1.91	10.46	1.75	7.93
7	116.11	78	2.3	5.53	1.49	8.63	2.78	6.26
8	85.96	67	1.98	6.7	1.73	8.27	1.87	12.15
9	78.47	54	1.86	6.11	1.93	4.78	2.62	9.64
10	80.61	67	1.84	12.2	1.41	5.4	2.08	8.75
11	21.28	62	2.34	2.93	2.06	6.79	2.17	9.83
12	82.62	60	2.54	8.24	2.35	9.94	2.03	5.99
13	0.2	18	2.49	6.9	2.33	4.88	1.64	9.36
14	0	13	1.89	9.54	2.29	6.78	1.52	7.04
15	1.2	20	2.29	8.12	1.74	6.44	1.44	7.81
16	4.17	24	1.78	3.78	1.6	1.17	1.14	12.04
17	0	29	1.49	8.03	2.3	6.05	2.07	10.62
18	0	20	2.63	9.2	2.32	4.23	1.63	11.17
19	73.94	56	2.05	9.88	2.32	4.89	2.52	3.04
20	85.02	44	1.52	14.76	2.68	1.71	2.66	6.47

Conclusions

This study is the first test on the simultaneous use of two bio-inspired algorithms in the field of architecture. As depicted by the case studies, the combined model developed by the study is effective. In other words, the two bio-inspired algorithms, the swarm intelligence algorithm and GA, could work together. Specifically, the GA optimized swarm agents in motion.

In this study, the parcel boundary and the maximum achievable height of the mass were the constraints in the swarm simulation, and the maximization of the average surface area of the Delaunay/Voronoi mesh associated with the alignment, cohesion, and separation values was defined as the criterion (objective). Accordingly, the swarm simulation was optimized using the Galapagos program to achieve the desired values. As this study focused on the effective use of two bio-inspired algorithms together, other architectural parameters (e.g., functional, structural, and material parameters) were excluded from the study's scope, with the exception of increasing the surface area parameter, which was determined as an objective.

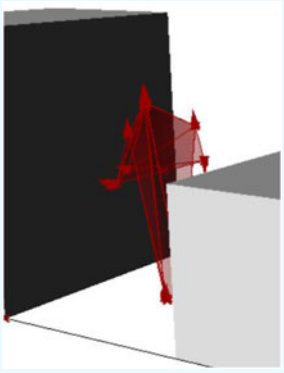
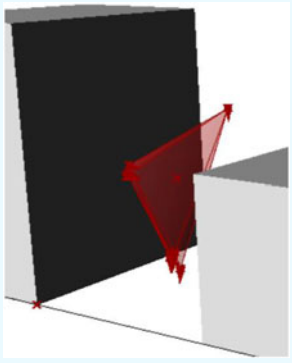
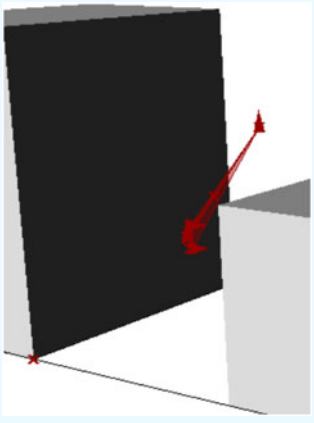
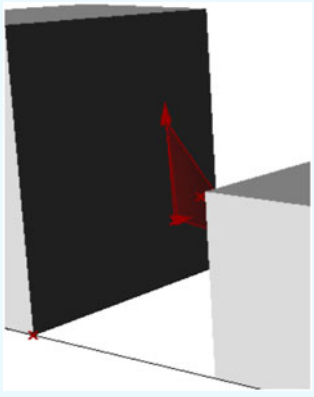
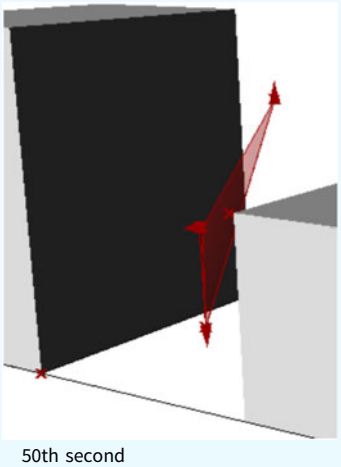
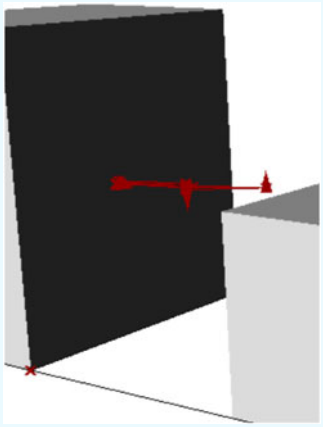
The ability of two bio-inspired algorithms to run together is the main inference of this study. However, based on the results obtained in this study, the following additional inferences are open for discussion.

Normally, the rule-based swarm algorithm does not work as an optimization algorithm. As many combinations of alignment, cohesion, and separation values are possible in the rule-based swarm intelligence approach, it is very difficult to identify maximum or minimum values of specific criteria and constraints without using an optimization tool. Therefore, the benefit of the combined method produced in this study is that the combination values of self-organizing swarm agents can be found through optimization (with the GA).

Today, the concept design process has changed from a two-dimensional approach (e.g., manual sketches on paper) to a static three-dimensional approach (e.g., digital sketches on computer-aided design programs). Accordingly, concept design with intelligent modeling and simulation tools has started superseding other, less advanced, methods. However, computer programs can sometimes restrict the flexibility of the designer, as they do not facilitate sufficient form formation during the conceptual design phase. Therefore, as suggested by this study, various new methods, which can be used in computer-aided design platforms, should be developed. In this study, the method, which involved the simultaneous use of two bio-inspired approaches – the swarm intelligence algorithm and GA – by utilizing advanced technologies, enabled the designer to increase the speed of the sketching process in the conceptual design stage of architectural design. The digital sketches, which were formed in line with the established criteria and constraints throughout the simulation, could introduce new ideas to the designer, which may inspire the designer to initiate a process of rediscovery. As demonstrated in the case studies (Delaunay and Voronoi), optimization offers

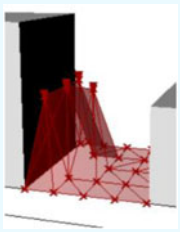
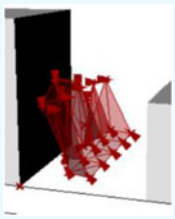
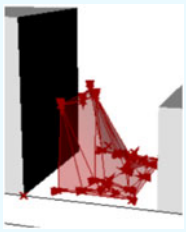
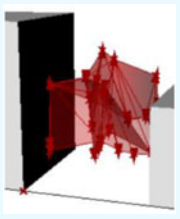
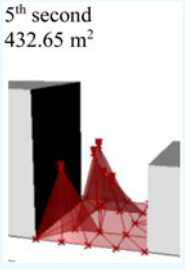
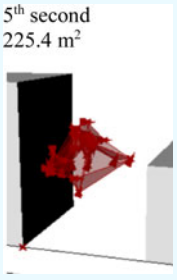
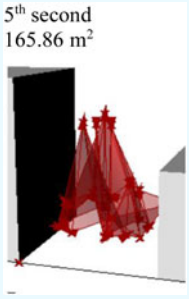
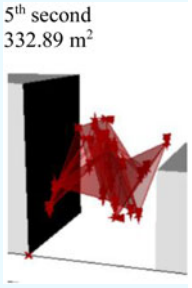
https://doi.org/10.1017/S0890060420000323 Published online by Cambridge University Press

Table 6. Sections of the swarm simulation initiated by the values achieved from the third Galapagos optimization for the Delaunay mesh area (minimizing the surface area)

Initial setup	Alignment steer force: 2.03 unit, alignment max distance: 5.99 m
	Cohesion steer force: 2.21 unit, cohesion max distance: 5.97 m
	Separation Steer Force: 1.75 unit, separation max distance: 5.97 m
	Avoidance parameter: 2 points (radius of avoidance distance: 9 m, steer force: 1 unit)
	Number of agents: 25 (X:5, Y:5)
	Box: 12 × 16 × 12 m
Swarm Simulation results	 10th second Delaunay mesh area: 71.31 m^2
	 20th second Delaunay mesh area:40.55 m^2
	 30th second Delaunay mesh area:15.76 m^2
	 40th second Delaunay mesh area: 34.5 m^2
	 50th second Delaunay mesh area: 36.73 m^2
	 60th second Delaunay mesh area: 23.87 m^2

https://doi.org/10.1017/S0890060420000323 Published online by Cambridge University Press

Table 7. Analysis of the change of the surface area according to the alignment, cohesion, and separation values

A SF:1 A MD:2 C SF:1 C MD:2 S SF:1 S MD:2	 5th second 432.65 m^2	A SF: 1 **A MD:5** C SF:1 C MD:2 S SF: 1 S MD:2	 5th second 225.4 m^2	A SF:1 A MD:2 C SF:1 **C MD:5** S SF: 1 S MD:2	 5th second 165.86 m^2	A SF: 1 A MD:2 C SF:1 C MD:2 S SF: 1 **S MD:5**	 5th second 332.89 m^2
	 10th second 375.69 m^2		 10th second 143.47 m^2		 10th second 317.57 m^2		 10th second 296.98 m^2

A SF: Alignment steer force value, A MD: Alignment maximum distance value (m), C SF: Cohesion steer force value, C MD: Cohesion maximum distance value (m), S SF: Separation steer force value, S MD: Separation maximum distance value (m)

various form alternatives (in accordance with the established objectives and constraints) to the designer and thereby, strengthens the conceptual design stage.

Limitations and future work

As the main focus of this study was based on the simultaneous use of two algorithms, an experiment was carried out on a simple objective. However, using more sophisticated objectives, experiments should be done on the interoperability of these two algorithms in the field of architecture and design.

In the combined method, Galapagos optimization was initiated without a runtime limit. However, the number of generations produced was observed to differ in each test. Consequently, the computation times were also different for each test. Another observation was that the computation time varied according to the objective of minimizing or maximizing the surface area of the form and according to the type of form being used. Moreover, the number of swarm agents and the width of the area that the swarm agents effect, has a significant impact on the computation time. The type of computer hardware was also found to influence the computation time. It will therefore be advantageous to perform various tests and comparative analysis on the aforementioned topic.

In this paper, if we make inferences in line with the objective chosen within the scope of optimization, the optimization of boids does not provide direct results (within the scope of Voronoi and Delaunay geometries). The results demonstrate that the optimal parameter values for alignment, cohesion, and separation can vary, and the run time of the swarm simulation affects the resulting geometry. However, GA optimization provides a conclusion about the overall value range, and this could be an answer to another research question within the scope of this study: "Can digital pathways be developed to provide more flexible solutions in the conceptual design phase?" The method used in this study may be suitable for the conceptual design phase in the digital environment because this method contributes to the sketching process in the digital environment with different alternatives offered to the designer, and thus, the designer can revise and reshape their ideas. This conforms to the definition of *ill-structured* design options that should take place in the conceptual design phase.

As a future direction of this study, the maximization of the surface area using a multi-objective optimization add-on (e.g., the Octopus add-on) should be optimized using another parameter. Some examples of this other parameter are the provision of daylight control using the Honeybee add-on, the generation of environmentally conscious architectural designs using the Ladybug add-on, the provision of acoustic control using the Pachyderm add-on, and the provision of structure-related controls using the Karamba add-on.

References

Ab Rashid M, Tiwari A and Hutabarat W (2019) Integrated optimization of mixed-model assembly sequence planning and line balancing using multi-objective discrete particle swarm optimization. *Artificial Intelligence for Engineering Design, Analysis and Manufacturing*. doi:10.1017/S0890060419000131

Adeniran A and El Ferik S (2017) A reinforced combinatorial particle swarm optimization based multimodel identification of nonlinear systems. *Artificial Intelligence for Engineering Design, Analysis and Manufacturing* **31**, 327–358.

Agirbas A (2018) Performance-based design optimization for minimal surface based form. *Architectural Science Review* **61**, 384–399.

Ahlquist S, Erb D and Menges A (2015) Evolutionary structural and spatial adaptation of topologically differentiated tensile systems in architectural

https://doi.org/10.1017/S0890060420000323 Published online by Cambridge University Press

design. *Artificial Intelligence for Engineering Design, Analysis and Manufacturing* **29**, 393–415.

Alaliyat S, Yndestad H and Sanfilippo F (2014) Optimisation of boids swarm model based on genetic algorithm and particle swarm optimisation algorithm (comparative study). *Proceedings of 28th European Conference on Modelling and Simulation (ECMS)*, Brescia, Italy, May 27–30.

Attia S, Gratia E, De Herde A and Hensen JL (2012) Simulation-based decision support tool for early stages of zero-energy building design. *Energy and Buildings* **49**, 2–15.

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

Bentley PJ and Wakefield JP (1997) Conceptual evolutionary design by a genetic algorithm. *Engineering Design and Automation* **3**, 119–132.

Benvenuti S, Ceccanti F and De Kestelier X (2013) Living on the moon: topological optimization of a 3D-printed lunar shelter. *Nexus Network Journal* **15**, 285–302.

Benyus JM (1998) *Biomimicry: Innovation Inspired by Nature*. New York: Harper Collins Publishers.

Bonabeau E, Dorigo M and Theraulaz G (1999) *Swarm Intelligence. From Natural to Artificial Systems*. New York: Oxford University Press.

Camazine S (1991) Self-organizing pattern-formation on the combs of honeybee colonies. *Behavioral Ecology and Sociobiology* **28**, 61–76.

Camazine S, Deneubourg JL, Franks NR, Sneyd J, Theraulaz G and Bonabeau E (2001) *Self-Organization in Biological Systems*. New Jersey: Princeton University Press.

Cerver tools (2018) Locust – Behavioral Animation Tools. Available at https://www.grasshopper3d.com/group/cervertools (accessed 31 March 2018).

Chakrabarti A and Shu L (2010) Biologically inspired design. *Artificial Intelligence for Engineering Design, Analysis and Manufacturing* **24**, 453–454.

Chakrabarti A, Sarkar P, Leelavathamma B and Nataraju B (2005) A functional representation for aiding biomimetic and artificial inspiration of new ideas. *Artificial Intelligence for Engineering Design, Analysis and Manufacturing* **19**, 113–132.

Chatzi EN and Koumousis VK (2009) Optimal inelastic design of multistorey reinforced concrete buildings towards uniform distribution of minimal damage. *International Conference on Structural Engineering Dynamics*, Ericeira, Portugal, June 22–24.

Chen Y-W, Kobayashi K, Huang X and Nakao Z (2006) Genetic algorithms for optimization of boids model. In Gabrys B, Howlett RJ and Jain LC (eds), *Knowledge-Based Intelligent Information and Engineering Systems*. Berlin: Springer, pp. 55–62.

Cui Z and Shi Z (2009) Boid particle swarm optimisation. *International Journal of Innovative Computing and Applications* **2**, 77–85.

DeLanda M (2002) Deleuze and the use of the genetic algorithm in architecture. *Architectural Design* **71**, 9–12.

Do E (2002) Drawing marks, acts, and reacts: toward a computational sketching interface for architectural design. *Artificial Intelligence for Engineering Design, Analysis and Manufacturing* **16**, 149–171.

Dorigo M and Stutzle T (2004) *Ant Colony Optimization*. Cambridge, MA: The MIT Press.

Dorigo M, Bonabeau E and Theraulaz G (2000) Ant algorithms and stigmergy. *Future Generation Computer Systems* **16**, 851–871.

Eberhart RC and Kennedy J (1995) A new optimizer using particle swarm theory. *Proceedings of Sixth International Symposium on Micro Machine and Human Science*, Nagoya, Japan, October 4–6, pp. 39–43.

Eberhart RC, Dobbins RW and Simpson PK (1996) *Computational Intelligence PC Tools*. Boston: Academic Press.

Elbeltagi E, Hegazy T and Grierson D (2005) Comparison among five evolutionary-based optimization algorithms. *Advanced Engineering Informatics* **19**, 43–53.

Evins R (2013) A review of computational optimization methods applied to sustainable building design. *Renewable and Sustainable Energy Reviews* **22**, 230–245.

Felkner J, Chatzi E and Kotnik T (2013) Interactive particle swarm optimization for the architectural design of truss structures. *Proceedings of the 2013 IEEE Symposium Computational Intelligence for Engineering Solutions (CIES)*, Singapore, April 16–19, pp. 15–22.

Fourie PC and Groenwold AA (2002) The particle swarm optimization algorithm in size and shape optimization. *Structural and Multidisciplinary Optimization* **23**, 259–267.

Frazer JH (1995) *An Evolutionary Architecture*. London: AA Publications.

Garner S (1992) The undervalued role of drawing in design. In Thistlewood D (ed), *Drawing Research and Development*. London: Longman, pp. 98–109.

Goel V (1992) Ill-structured representations for Ill-structured problems. *Proceedings of the Fourteenth Annual Conference of the Cognitive Science Society*, Bloomington, Indiana, July 29–August 1, pp. 130–135.

Goel V (1995) *Sketches of Thought*. Cambridge, MA: MIT Press.

Goldberg DE and Holland JH (1988) Genetic algorithms and machine learning. *Machine Learning* **3**, 95–99.

González J and Fiorito F (2015) Daylight design of office buildings: optimisation of external solar shadings by using combined simulation methods. *Buildings* **5**, 560–580.

Hamdy M, Palonen M and Hasan A (2012) Implementation of pareto-archive NSGA-II algorithms to a nearly-zero-energy building optimisation problem. *Proceedings of the First Building Simulation and Optimization Conference*, Loughborough, UK, September 10–11, pp.181–188.

Hofmeyer H and Davila Delgado J (2015) Coevolutionary and genetic algorithm based building spatial and structural design. *Artificial Intelligence for Engineering Design, Analysis and Manufacturing* **29**, 351–370.

Hu Y-J, Wang Y, Wang Z-L, Wang Y-Q and Zhang BC (2014) Machining scheme selection based on a new discrete particle swarm optimization and analytic hierarchy process. *Artificial Intelligence for Engineering Design, Analysis and Manufacturing* **28**, 71–82.

Jin Y and Benami O (2010) Creative patterns and stimulation in conceptual design. *Artificial Intelligence for Engineering Design, Analysis and Manufacturing* **24**, 191–209.

Jo JH and Gero JS (2006) Space layout planning using an evolutionary approach. *Artificial Intelligence in Engineering* **12**, 149–162.

Kämpf J, Wetter M and Robinson D (2010) A comparison of global optimization algorithms with standard benchmark functions and real-world applications using EnergyPlus. *Journal of Building Performance Simulation* **3**, 103–120.

Kennedy J and Eberhart R (1995) Particle swarm optimization. *Proceedings of ICNN'39 – International Conference on Neural Networks*, Perth, Australia, November 27–December 1, pp. 1942–1948.

Kim S (2013) Interval estimation of construction cost using case-based reasoning and genetic algorithms. *Journal of Asian Architecture and Building Engineering* **11**, 327–334.

Kramar D, Cica D, Sredanovic B and Kopac J (2015) Design of fuzzy expert system for predicting of surface roughness in high-pressure jet assisted turning using bioinspired algorithms. *Artificial Intelligence for Engineering Design, Analysis and Manufacturing* **30**, 96–106.

Lee DT and Schachter BJ (1980) Two algorithms for constructing a Delaunay triangulation. *International Journal of Computer Information Sciences* **9**, 219–242.

Lipson H and Shpitalni M (2000) Conceptual design and analysis by sketching. *Artificial Intelligence for Engineering Design, Analysis and Manufacturing* **14**, 391–401.

Machairas V, Tsangrassoulis A and Axarli K (2014) Algorithms for optimization of building design: a review. *Renewable and Sustainable Energy Reviews* **31**, 101–112.

Mukerjee A, Agrawal R, Tiwari N and Hasan N (1997) Qualitative sketch optimization. *Artificial Intelligence for Engineering Design, Analysis and Manufacturing* **11**, 311–323.

Nguyen A, Reiter S and Rigo P (2014) A review on simulation-based optimization methods applied to building performance analysis. *Applied Energy* **113**, 1043–1058.

Partridge BL (1982) The structure and function of fish schools. *Scientific American* **246**, 114–123.

Pawlyn M (2011) *Biomimicry in Architecture*. London: RIBA Publishing.

Pedersen Zari M (2015) Ecosystem processes for biomimetic architectural and urban design. *Architectural Science Review* **58**, 106–119.

https://doi.org/10.1017/S0890060420000323 Published online by Cambridge University Press

Rajan S (1995) Sizing, shape, and topology design optimization of trusses using genetic algorithm. *Journal of Structural Engineering* **121**, 1480–1487.

Rasheed K and Hirsh H (1999) Learning to be selective in genetic-algorithm-based design optimization. *Artificial Intelligence for Engineering Design, Analysis and Manufacturing* **13**, 157–169.

Rawat CD, Shahani A, Natu N, Badami A and Hingorani R (2012) A genetic algorithm for VLSI floor planning. *International Journal of Engineering Science and Advanced Technology* **2**, 412–415.

Reddy MJ and Kumar DN (2007) An efficient multi-objective optimization algorithm based on swarm intelligence for engineering design. *Engineering Optimization* **39**, 49–68.

Renner G and Ekárt A (2003) Genetic algorithms in computer aided design. *Computer Aided Design* **35**, 709–726.

Reynolds CW (1987) Flocks, herds, and schools: a distributed behavioral model. *Computer Graphics* **21**, 25–34.

Reynolds, C. (2017) Boids: Background and Update. Available at http://www.red3d.com/cwr/boids/ (accessed 18 August 2017).

Rutten D (2013) Galapagos: on the logic and limitations of generic solvers. *Architectural Design* **83**, 132–135.

Sartori J, Pal U and Chakrabarti A (2010) A methodology for supporting "transfer" in biomimetic design. *Artificial Intelligence for Engineering Design, Analysis and Manufacturing* **24**, 483–506.

Schon DA (1983) *The Reflective Practitioner: How Professionals Think in Action*. New York: Basic Books.

Schon DA and Wiggins G (1992) Kinds of seeing and their functions in designing. *Design Studies* **13**, 135–156.

Shi X, Tian Z, Chen W, Si B and Jin X (2016) A review on building energy efficient design optimization from the perspective of architects. *Renewable and Sustainable Energy Reviews* **65**, 872–884.

Su Z and Yan W (2015) A fast genetic algorithm for solving architectural design optimization problems. *Artificial Intelligence for Engineering Design, Analysis and Manufacturing* **29**, 457–469.

Suwa M and Tversky B (1996) What architects see in their sketches: implications for design tools. In *Proceedings of CHI'96*, Vancouver, BC, Canada, April 13–18, pp. 191–192.

Tedeschi A (2014) *AAD_Algorithms-Aided Design: Parametric Strategies Using Grasshopper*. Brienza: Le Penseur Publisher.

Theraulaz G and Bonabeau F (1995) Modelling the collective building of complex architectures in social insects with lattice swarms. *Journal of Theoretical Biology* **177**, 381–400.

Theraulaz G, Gautrais J, Camazine S and Deneubourg J (2003) The formation of spatial patterns in social insects: from simple behaviours to complex structures. *Philosophical Transactions: Mathematical, Physical and Engineering Sciences* **361**, 1263–1282.

Tuhus-Dubrow D and Krarti M (2009) Comparative analysis of optimization approaches to design building envelope for residential buildings. *ASHRAE Transactions* **115**, 554.

van Embden Andres MV, Turrin M and von Buelow P (2011) Architectural DNA: a genetic exploration of complex structures. *International Journal of Architectural Computing* **9**, 133–149.

Vattam S, Helms M and Goel A (2010) A content account of creative analogies in biologically inspired design. *Artificial Intelligence for Engineering Design, Analysis and Manufacturing* **24**, 467–481.

Vincent J, Bogatyreva O, Bogatyrev N, Bowyer N and Pahl K (2006) Biomimetics: its practice and theory. *Journal of the Royal Society Interface* **3**, 471–482.

Wang W, Zmeureanu R and Rivard H (2005) Applying multi-objective genetic algorithms in green building design optimization. *Building and Environment* **40**, 1512–1525.

Wetter M and Wright J (2004) A comparison of deterministic and probabilistic optimization algorithms for non-smooth simulation-based optimization. *Building and Environment* **39**, 989–999.

Wright J and Alajmi A (2005) The robustness of genetic algorithms in solving unconstrained building optimization problems. *Proceedings of the Ninth International IBPSA Conference*, Montreal, Canada, August 15–18, pp.1361–1368.

Yu S, Austern G, Jirathiyut T and Moral M (2014) Climatic formations: evolutionary dynamics of urban morphologies. *Journal of Asian Architecture and Building Engineering* **13**, 317–324.

Yu R, Gu N, Ostwald M and Gero J (2015) Empirical support for problem–solution coevolution in a parametric design environment. *Artificial Intelligence for Engineering Design, Analysis and Manufacturing* **29**, 33–44.

Asli Agirbas is an Associate Professor in the Department of Architecture at Fatih Sultan Mehmet Vakif University. She received her MS Arch from Pratt Institute, New York and her PhD from Mimar Sinan Fine Arts University, Istanbul. Her recent research interest is focused on the use of swarm intelligence in architectural design. She also published articles about parametric design, CAD education, and macro-scale designs and optimization in the field of architecture.

https://doi.org/10.1017/S0890060420000323 Published online by Cambridge University Press