Diagrid Façade Design for Public Pool Building Using Differential Evolution

Ünlü, Püren; Ekici, Berk; Chatzikonstantinou, Ioannis; Sariyildiz, Sevil; Tasgetiren, Mehmet Fatih; Çubukçuoglu, Cemre

Publication date
2018

Document Version
Final published version

Published in
Proceedings of the 12th International Symposium on Tools and Methods of Competitive Engineering (TMCE 2018)

Proceedings of TMCE 2018, 7-11 May, 2018, Las Palmas de Gran Canaria, Spain
Edited by: I. Horváth, J.P. Suárez Rivero and P.M. Hernández Castellano

DIAGRID FAÇADE DESIGN FOR PUBLIC POOL BUILDING USING DIFFERENTIAL EVOLUTION

Nezahat Puren Unlu
Department of Architecture, Yasar University, Izmir, Turkey
npurenunlu@hotmail.com

Berk Ekici
Ioannis Chatzikonstantinou
I. Sevil Sariyildiz
Department of Architecture, Yasar University, Izmir, Turkey
Chair of Design Informatics, TU Delft, Delft, The Netherlands
{berk.ekici, sevil.sariyildiz } @yasar.edu.tr
{B.Ekici-1, I.Chatzikonstantinou, I.S.Sariyildiz} @tudelft.nl

M. Fatih Tasgetiren
Department of International Logistics
Yasar University, Izmir, Turkey
fatih.tasgetiren@yasar.edu.tr

Cemre Cubukcuoglu
Department of Interior Architecture and Environmental Design, Yasar University, Izmir, Turkey
Chair of Design Informatics, TU Delft, Delft, The Netherlands
cemre.cubukcuoglu@yasar.edu.tr
C.Cubukcuoglu@tudelft.nl

ABSTRACT

This paper discusses a set of façade design alternatives for form-finding problem focusing on conceptual phase. In this respect, the aim of the research is to propose a multi-objective optimization approach for a façade design of public pool building. We present a set of solutions belonging to Self-adaptive Multi-objective Ensemble Differential Evolution (JE_DEMO) and Self-adaptive Multi-Objective Differential Evolution (JDEMO) algorithm. We focus on maximization of daylight performance and minimization of structural displacement. Based on results, two algorithms presented competitive results. Contributions are presented based on objectives functions as new trade-offs and proposed JE_DEMO algorithm for design problems.

KEYWORDS

Diagrid façade design; performance-based design; computational design; multi-objective optimization; differential evolution.

1. INTRODUCTION

Building envelopes are complex systems involving multiple requirements in the sense of architecture and engineering. In addition, these requirements contain numerous objectives that are conflicting in most of the cases.

In the field of architecture, designing façade is one of the most complex tasks. At this point, the role of the architect is not only facing with design concerns, but also to present a well performed daylight solution at the final process. This statement can be supported by some reasons:

- Daylight, which provides the natural lighting, is basic requirement for human being's daily life (Li & Tsang, 2008) ,
- Performance based design may support to reduce building's artificial lighting usage, as well as enhance indoor climate (Chatzikonstantinou, Ekici, Sarıyıldız, & Koyunbaba, 2015).

Based on these statements, it is possible to say that openings of buildings need much attention for daylight performance.

In the literature, there are several works on façade design related to our research. In this study (Kim, 2012), authors are considered the optimization for daylight performance in the façade system. The computational optimization produces alternatives in terms of opening forms. In the literature (Chatzikonstantinou et al., 2015), authors are attempted to identify façade configurations, including elements of the glass panel frames and construction materials. The overall purpose of the study is to maximize daylight performance of the building. Several scenarios are formulated in the manner of multi-objective optimization. Based on this study, it is possible to say that façade design has a significant effect on both daylight distribution and structural performance (Chatzikonstantinou et al., 2015; Perera & Sirimanna, 2014). This argument can be discussed over an example. For instance, while increasing the façade division count with bigger size of profiles, structural performance present satisfactory results. However, this fact causes to block natural daylight distribution for interior space. For this reason, daylight and structural performance are two conflicting objectives in the design task of building envelope.

Diagrid, as a façade pattern, is one of the most preferred design elements for building's envelope. According to (Leonard, 2007), diagonal elements created to the pattern of diagrid. Regarding to preferring diagrid form as façade element, numerous benefits can be shown as follows (Panchal & Patel, 2014):

- Reducing displacement,
- Requiring less structural elements,
- Increasing structural resistance and efficiency,
- So, use of materials more efficiently.

In this study, we deal with the design process of diagrid façade for public pool building as multi-objective optimization problem. The objectives are to maximize daylight performance (PDF) and minimize the displacement of the façade structure (v) with the aim of finding near-optimal design scenarios. JE_DEMO and JDEMO algorithms, which are based on differential evolution (DE) optimization algorithm, are used in this research.

The remaining paper is outlined as follows: Section 2 introduces the façade model. Section 3 presents the problem definition with the details of explanation façade model. Section 4 shows optimization algorithms. The computational optimization results over the problem are argued in Section 5. Finally, Section 6 concludes the study.

2. PARAMETRIC FAÇADE MODEL

Our scenario focuses on a parametric design of a pool building for the public use located in Bornova, Izmir, Turkey, having 38°27'N latitude, 27°12'E longitude and 17m altitude. While designing a building, location properties become significant to handle actual natural lighting data. Izmir is approximately whole days of year sunny so it can be an advantage for architects to improve daylight performance of a building in Izmir. Specifically, project site of this study is also suitable for making use of daylight. The problem entails the decisions regarding the design of façade elements and frames.

We consider daylight and structural objectives, which are conflicting. To realize this complex study, we solve the problem dividing into three phases, which are generation of shape, performance evaluation, and optimization for identifying façade configurations. To do this, required notations with explanations are stated in Table I.

The building in our case study is including pool area, service functions (i.e. changing rooms, rest rooms, cleaning rooms, and so forth), café for public usage, and the main entrance area as shown in Fig. 1. To realize this research, we focus on the design of pool unit.

In addition, this building includes two different pools for both adults and children. The dimensions of the one designed for adults are 25 m by 20 m, while the dimensions of one for children usage are 10 m by 15 m. The floor height of the pool is as decision variables.

Table 1 Notations and Explanations of Façade Model

Notations and Explanations of Façade Model	
Not.	***Exp.***
d_v	Count of vertical division of diagrid façade
d_h	Count of horizontal division of diagrid façade
b_i	Inner frame of base section
b_o	Outer frame of base section
h_i	Inner frame of height section
h_o	Outer frame of height section
th_i	Inner frame thickness
th_o	Outer frame thickness
w_{if}	Inner frame width
w_{of}	Outer frame width
h_r	Right front height of diagrid façade
h_l	Left front height of diagrid façade

The diagrid façade, which is located to the south part of the building, is composed of double pane glazing. Since façade design cannot be divided into just diagrids, sub-elements of triangles are generated. So that, combination of quad and triangle elements reveals the whole diagrid form. After generating the main façade design, frame is supported through the connection points located around the building's skin. The number of support points depends on the number of façade elements, which are made of steel material. When division count of façade design increase, count of support points are increasing. A detailed explanation of the structure components is presented in Fig. 2.

The main purpose of this study is to minimize v, and to maximize PDF. These objectives, which are conflicting, have an important role on the façade configuration. In addition, two constraints are defined in order to discover façade alternatives in acceptable margins. The first constraint function is focusing on the value of DF in order to reduce usage of artificial lighting during day hours. The second constraint is related with v in order to present well-performed façade alternatives in terms of structural performance.

Regarding the decision variables of the problem, we take into account dimensions of façade elements, profile sections, and floor height. Especially, in order to keep diversity of design solution in search space, façade elements are constituted as two different decision variable group. These are main (outer) frame variables and sub (inner) frame variables. An example of final design after these steps is illustrated in Fig.3.

3. PROBLEM DEFINITION

3.1. Objective Functions

In this research, the first objective function is formulated as to minimize v and the other objective is formulated as to maximize PDF as follows:

$$\text{Min}\left(v, \frac{1}{PDF}\right) \quad (1)$$

Subject to:

$$PDF \geq 2 \quad (2)$$

$$v \leq 0.05m \quad (3)$$

where PDF refers to performance of daylight for specific measurement points. Notations, types, ranges, and units of considered decision variables, are shown in Table II.

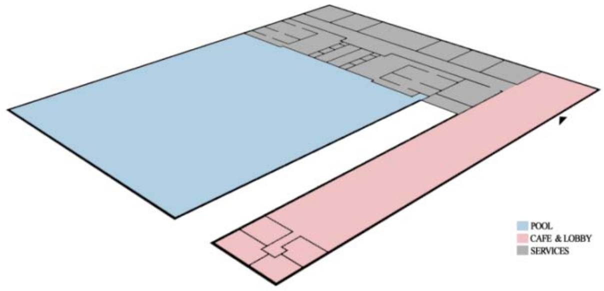

Figure 1 Building Plan Diagram.

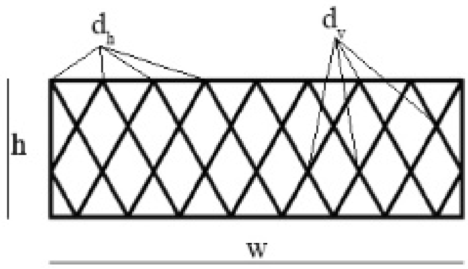

Figure 2 Diagram Façade Components

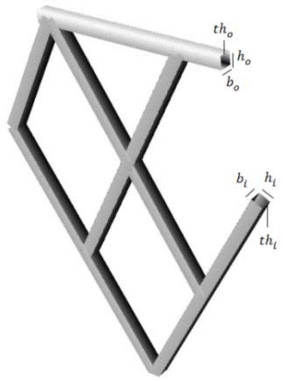

Figure 3 Diagrid Façade Construction Diagram

Table 2 Decision Variables of the Model

Decision Variables of the Model			
Notation	***Type***	***Range***	***Unit***
d_v	Integer	[3, 7]	Count
d_h	Integer	[6, 16]	Count
h_r	Real	[6, 15]	meter
h_f	Real	[6, 15]	meter
b_i	Real	[5, 50]	centimetre
b_o	Real	[5, 50]	centimetre
h_i	Real	[5, 50]	centimetre
h_0	Real	[5, 50]	centimetre
th_i	Real	[0.10, 0.30]	centimetre
th_0	Real	[0.10, 0.30]	centimetre

3.2. Daylight Objective

DF is the ratio of the illuminance level between inside (E_{inside}) and the outside ($E_{outside}$) of the indoor space under overcast sky at a specific point (Waldram, 1925). We determine 19 measurement points on the floor to calculate DF for each point in the pool area that is shown in Fig. 4. In our case, each measurement point's DF is combined in order to reach PDF objective function as follows:

$$PDF = Avg\left(DF_{Pt_{1,..,19}}\right) \quad (4)$$

where

$$DF = \frac{E_{inside}}{E_{outside}} . 100\% \quad (5)$$

According to LEED criteria, less than 2% DF for 75% of overall measuring points causes usage of artificial lighting during the day hours (Moon & Spencer, 1942). Related to this fact extra electrical equipment consumption is needed. Based on this reason constraint function expressed in equation (2). To employ DF calculation during the optimization process, a plug-in called DIVA (Jakubiec & Reinhart, 2011) is utilized.

3.3. Displacement Objective

In order to find the equilibrium state of the diagrid façade design, displacement (v) of the structural elements is considered. v is calculated by the help of Karamba 3D (Preisinger et al., 2018) plugin, which works in Grasshopper 3D (McNeel, 2013) environment, as follows:

$$F = K \times v \quad (6)$$

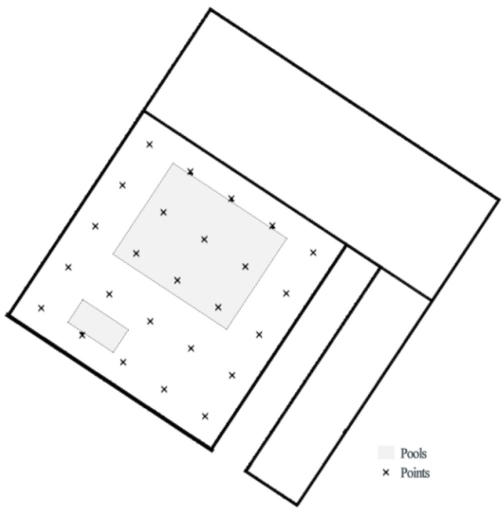

Figure 4 DF points on the floor in the pool area.

where: (*K*) is the stiffness of the diagrid façade structure and (*F*) is the loading force. From the point of load combination of two categories, namely dead loads and wind loads are situated. In accordance with equation (3), structural performance of each generated solution during the optimization process is kept within safety margins. From the standpoint of load combination, permanent actions (G_k) are taking into consideration for the buildings. The weight of the structure and all architectural components of the building can be shown as main components of G_k. As another action on buildings, variable actions (Q_k) can be mentioned (i.e. the wind pressure, snow weight). According to TS EN 1990:2002, the general formula of the load combination is as follows:

$$\sum_{f(x)} \gamma_G . G_{k.j} + \Upsilon_{Q.I} = Q_{k.I} + \sum_{f(x)} \gamma_{Q.I} . \Psi_{0.I} . Q_{k.i} \quad (7)$$

where γ_G states the partial factor for permanent action, γ_Q presents the partial factor for variable action and, Ψ_0 is the combination factor, $Q_{k.I}$ represents the leading variable action and the other variable actions. Based on this statement, following load combination is considered to realize this study as follows:

$$F = 1.35DL + W \quad (8)$$

where DL presents the dead load, and W states wind load of each generated alternative during the evolutionary computation process. To calculate DL as first step, following equation is implemented to generative model:

$$DL = \sum_{i=1}^{n} W_{str_i} \quad (9)$$

where W_{str_i} corresponds the weight of ith structural elements in the diagrid façade frame. Summation of each weight of structural element starting from i = 1 to n gives the total DL for each alternative. From the point of W, Eurocode standards (Cook, 2007) which specifies structural requirements within the European Union, is considered in this study. The basic wind velocity in the case region is considered as follows:

$$V_b = C_{dir} . C_{season} . V_{b,0} \quad (10)$$

where V_b corresponds the basic wind velocity, C_{dir} states the directional factor and, C_{season} matches the seasonal factor, and $V_{b,0}$ is the fundamental value of basic wind velocity. In accordance with the TS EN 1991-1-4:2007, C_{dir} and C_{season} is assumed as 1. Since the case study is located in Izmir, Turkey, $V_{b,0}$ is supposed as 26 m/s. Based on equation (10), V_b is determined as 26 m/s at the end. To calculate basic wind pressure (q_p) and the mean wind velocity ($V_m(z)$), following equations are taken into consideration:

$$q_p = \frac{p}{2}.\ V_b{}^2 \tag{11}$$

$$V_m(z) = \ c_r(z).\, c_o(z).\, v_b \tag{12}$$

where p is the density of air considered as 1.25kg/m3, $c_r(z)$ states the roughness factor and, $c_o(z)$ corresponds the orography factor that is usually taken as 1.0. To determine $c_r(z)$ at height z, equation (13), (14), and (15) are considered as follows:

$$c_r(z) = \ k_r.\, In\left(\frac{z}{z_0}\right) \quad for \quad z_{min} \leq z \leq z_{max} \tag{13}$$

$$c_r(z) = c_r(z_{min}) \quad for \quad z \leq z_{min} \tag{14}$$

$$k_r = 0.19\left(\frac{z}{z_{0,II}}\right)^{0,07} \tag{15}$$

where k_r states the terrain factor, z corresponds the height we consider for the study, z_0 represents the roughness length, z_{min} expresses the minimum height considered as 2.0m, z_{max} states the maximum height assumed as 200.0 m, and $z_{0,II}$ presents the terrain category II assumed as 0.05m. Once we calculate V_b, q_p, and $V_m(z)$ in accordance with the mentioned equations above, pressure of the peak wind velocity $[q_p(z)]$ at z height is calculated as follows:

$$q_p(z) = \ [1 + 7.\, I_v(z)].\, \frac{1}{2} p.\, V_m{}^2(z) \tag{16}$$

where $I_v(z)$, which represents turbulence intensity at z height, is defined by using equation (17), and (18) as follows:

$$I_v(z) = \frac{k_I}{c_o(z).In(\frac{z}{z_0})} \quad \text{for} \quad z_{min} \leq z \leq z_{max} \tag{17}$$

$$I_v(z) = I_v(z_{min}) \quad \text{for} \quad z \leq z_{min} \tag{18}$$

where k_I corresponds the turbulence factor that is assumed as 1. To calculate the wind force (F_w) on our case, as final step, following equation is computed for each generated solution:

$$F_w = c_s.\, c_d.\, c_f.\, q_p(z_e)\ .\ A_{ref} \tag{19}$$

where $c_s.c_d$ states the structural factor that is also considered as 1.0 in most of cases as a standard value, c_f is the force coefficient of the structure we focus on, $q_p(z_e)$ represents the peak velocity pressure at z_e height, and A_{ref} is the reference area that we implement our wind load force. Illustration of applied loads is shown in Fig. 5.

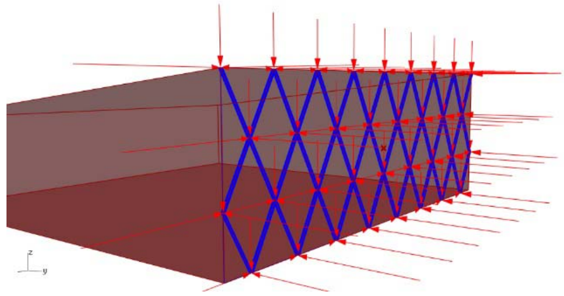

Figure 5 Load Combinations

4. OPTIMIZATION ALGORITHMS

Regarding the multi-objective optimization problems (MOPs), many researches have been published in the current literature. Through the years, Evolutionary Algorithms (EAs) have become one of the most common optimization algorithm type. In this respect, considering of EAs to MOPs are denoted as Multi-Objective Evolutionary Algorithms (MOEAs). In general, genetic operators regarding to EAs are employed to MOEAs for solving MOPs. With respect to implementation of these algorithms, benchmark and engineering problems are widely focused.

Differential Evolution (DE), which is proposed by (Price, Storn, & Lampinen, 2005; Storn & Price, 1995), is attracted much attention in the literature (Das & Suganthan, 2011; Mezura-Montes, Miranda-Varela, & del Carmen Gómez-Ramón, 2010). Based on this, we can say that satisfactory solutions for both unconstrained and constrained MOPs are presented by DE (Abbass, 2002; Abbass, Sarker, & Newton, 2001; Hernández-Díaz, Santana-Quintero, Coello Coello, Caballero, & Molina, 2006; Madavan & Biegel, 2002; Robič & Filipič, 2005; Santana-Quintero, Hernández-Díaz, Molina, Coello, & Caballero, 2010; Tušar & Filipič, 2007; Xue, Sanderson, & Graves, 2003; Zamuda, Brest, Boskovic, & Zumer, 2007).

During the evolutionary stage, mutation factor (F) and crossover rate (CR) play a vital role in DE. Concerning this topic, self-adaptive DE (JDE) is capable to update F and CR determining a certain of probability for each generation (Brest, 2009; Brest, Greiner, Boskovic, Mernik, & Zumer, 2006). The algorithm is turned out a very simple and efficient EA, with a small modification on the control parameters of DE. Based on JDE with mentioned modifications above, DEMOwSA is proposed (Zamuda et al., 2007) by using the properties of

DEMO for MOPs (Robič & Filipič, 2005; Tušar & Filipič, 2007).

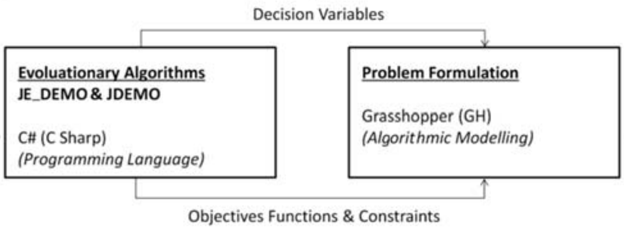

Figure 6 GH and DE algorithms interaction

As mentioned above, DE with multi-objective strategy, is one of the most popular EAs for MOPs. In this research paper, inspiring from DEMO with some modifications, JE_DEMO is proposed. To deal with problem on hand, JE_DEMO is compared to the JDEMO. Difference between two optimization methods mentioned in the following sections. The flow of the study is illustrated in Fig. 6.

4.1. Multi-objective Self Adaptive Differential Evolution (JDEMO)

DE is discussed by Das et al. (Das, Mullick, & Suganthan, 2016) in a review paper with its success considering the latest approaches developed. Pseudo code for single objective basic DE is shown in Fig. 7.

Begin
  $t = 0$
  Produce initial target population x_i^t $i = 1, .., N$
  Compute $f(x_i^t)$ for $i = 1, .., N$
  For $t = 1$ to MAXGEN **Do**
    For $i = 1$ to N **Do**
      Choose uniform random numbers
      $p \neq q \neq r \neq i \in (1, N)$:
      $k = rand(\,)\%D$
      For $j = 1$ to D **Do**
        If $rand_j[0,1) < CR$ or $j = k$ **Then**
          $u_i^{j,t+1} = x_p^{j,t} + F \times (x_q^{j,t} - x_r^{j,t})$
        Else
          $u_i^{j,t+1} = x_i^{j,t}$
        End If
      End For
      If $f(u_i^{t+1}) \leq f(x_i^t)$ **Then**
          $x_i^{t+1} = u_i^{t+1}$
      Else
          $x_i^{t+1} = x_i^t$
      End If
    End For
    $t = t + 1$
  End For
End

Figure 7 Pseudo code of DE/rand/1/bin

JDE, which is easy to implement on basic DE, is capable to converge much faster than the DE for high dimensional and complex problems. In the procedure of JDE, F_i and CR_i values are assigning for each individual. For the initial population, CR_i is taken as 0.5, while F_i is defined as 0.9. For further generations, each value is updated as follows:

$$F_i^{t+1} = \begin{cases} F_{min} + R_1.F_{max} & if\ R_2 < p_1 \\ F_i^t & otherwise \end{cases} \tag{20}$$

$$CR_i^{t+1} = \begin{cases} R_3 & if\ R_4 < p_2 \\ CR_i^t & otherwise \end{cases} \tag{21}$$

where $R_j \in \{1,2,3,4\}$ are random numbers, which are uniformly distributed between 0 and 1. Determining probability of self-adaptive procedure, we define p_1 and p_2 that are both taken as 0.1. On the other hand, F_{min} is defined as 0.1, while F_{max} is 0.9. In order to extend JDE described above to multi-objective constrained optimization problems, the combination of the target (x_i) and trial (u_i) populations is determined. Afterwards, we make use of non-dominated sorting (Deb, Pratap, Agarwal, & Meyarivan, 2002). The outline of the multi-objective JDE denoted as JDEMO is given in Fig. 8.

Step 1: Set $t = 0$ and create a random target population X_t for N size.
Step 2: Apply mutation and crossover strategies to X_t for getting U_t for N size.
Step 3: If the termination criteria is satisfied, stop and return X_t.
Step 4: Combine two populations as $R_t = X_t \cup U_t$
Step 5: Apply the fast non-dominated sorting procedure for R_t and define non-dominated fronts $f_1, f_2, .., f_k$ in R_t.
Step 6: For $i = 1, .., k$, do the following phases:
  Step 6.1: Calculate the crowding distance of solutions in f_i.
  Step 6.2: Create X_{t+1} as follows:
  • if $|X_{t+1}| + |f_i| \leq N$, then set $X_{t+1} = X_{t+1} \cup f_t$
  • if $|X_{t+1}| + |f_i| > N$, then add the least crowded $N - |X_{t+1}|$ solutions from f_i to X_{t+1}.
Step 7: Apply mutation and crossover operators to X_{t+1} and get U_{t+1} for N size.
Step 8: Set $t = t + 1$ and return to Step 3.

Figure 8 JDEMO Algorithm

4.2. Multi-objective Self-Adaptive Ensemble Differential Evolution (JE_DEMO)

In this paper, considering the previous explanations, JDEMO is extended with the ensemble approach inspiring from Tasgetiren et al. (Tasgetiren, Suganthan, Pan, Mallipeddi, & Sarman, 2010), Mallipeddi et al. (Mallipeddi, Suganthan, Pan, & Tasgetiren, 2011), and Das et al. (Das & Suganthan, 2011; Mezura-Montes et al., 2010). The main difference between JE_DEMO and DEMO is explained in the following lines. JDEMO employs only one mutation strategy, which is presented in Fig.6. With this approach, JDEMO benefits from three different individuals $(x_p^j, x_q^j, and\ x_r^j)$ to generate mutant population. This way of generating mutated individuals is totally different than genetic algorithm's approach. On the other hand, JE_DEMO uses several mutation strategies for each individuals. Therefore, three mutation strategies are employed to each individual to generate the mutant population. To do this, each decision variable has values pool for competition of producing better future offspring according to their success in the past generations. Following mutation strategies (M_i) are used in this research:

$$M_1: v_i^{j,t+1} = x_p^{j,t} + F\left(x_q^{j,t} - x_r^{j,t}\right) \tag{22}$$

$$\mathrm{M}_2: v_i^{j,t+1} = x_{best}^{j,t} + F\left(x_q^{j,t} - x_r^{j,t}\right) \tag{23}$$

$$M_3: v_i^{j,t+1} = x_i^{j,t} + F\left(x_{best}^{j,t} - x_i^{j,t}\right) + F\left(x_p^{j,t} - x_q^{j,t}\right) \tag{24}$$

where $\mathrm{p,q,r}$ are randomly selected individuals from the target population $\left(\mathrm{p} \neq \mathrm{q} \neq \mathrm{r} \neq \mathrm{i} \in (1,..,\mathrm{N})\right)$. Considering this, $\mathrm{j} = 1,..,\mathrm{D}$ and $\mathrm{F} > 0$. The outline JE_DEMO algorithm is given in Fig. 9.

Step 1: Set $\mathrm{t} = 0$ and $\mathrm{M_{max}} = 3$ and create a random target population $\mathrm{X_t}$ for N size.
Step 2: Assign a mutation strategy to each individual randomly
$\mathrm{M_i} = \mathrm{rand()\%M_{max}}$ for $\mathrm{i} = 1,..,\mathrm{N}$
Step 3: Apply F and CR parameters to $\mathrm{X_t}$ for getting $\mathrm{U_t}$ for N size.
Step 4: If the termination criteria is satisfied, stop and return $\mathrm{X_t}$.
Step 5: Combine two populations as $\mathrm{R_t} = \mathrm{X_t} \cup \mathrm{U_t}$
Step 6: Apply the fast non-dominated sorting procedure for $\mathrm{R_t}$ and define non-dominated fronts $\mathrm{f_1, f_2, \dots, f_k}$ in $\mathrm{R_t}$.
Step 7: For $\mathrm{i} = 1,..,\mathrm{k}$, do the following phases:
Step 7.1: Calculate the crowding distance of solutions in $\mathrm{f_i}$.
Step 7.2: Create $\mathrm{X_{t+1}}$ as follows:
- if $|\mathrm{X_{t+1}}| + |\mathrm{f_i}| \leq \mathrm{N}$, then set $\mathrm{X_{t+1}} = \mathrm{X_{t+1}} \cup \mathrm{f_t}$
- if $|\mathrm{X_{t+1}}| + |\mathrm{f_i}| > \mathrm{N}$, then add the least crowded $\mathrm{N} - |\mathrm{X_{t+1}}|$ solutions from $\mathrm{f_i}$ to $\mathrm{X_{t+1}}$.

Step 8: Apply mutation and crossover operators to $\mathrm{X_{t+1}}$ and get $\mathrm{U_{t+1}}$ for N size.
Step 9: Set $\mathrm{t} = \mathrm{t} + 1$ and return to Step 4.

Figure 9 JE_DEMO Algorithm

5. COMPUTATIONAL RESULTS AND DISCUSSION

As we mentioned before, two optimization algorithms, namely JE_DEMO and JDEMO are considered. For the optimization process, both runs are completed through a computer with Intel i5 core processor at 3.1 GHz, with 28 GB Ram Gskill dualkitx2 (1333 MHz) Kingston hyperx, and 480 GB solid-state drive. For both algorithms, population size is taken as 100. During the optimization, 10 generations took approximately 1.6-hour cpu time. Due to the lack of time, optimization process is stopped after 50 generations. Fig. 10. shows the Pareto chart solutions gained after this process.

For analysing the performance of algorithms, we discuss the hypervolume (HV) indicator that is calculating the volume of the non-dominated portion of the objective space (Bader & Zitzler, 2011). In order to calculate the HV for the JE_DEMO algorithm, we employed 42 non-dominated solutions whereas for JDEMO, we used 51 non-dominated solutions.

Comparing the optimization results from two different groups, PDF and v results are slightly different. The reference points are taken as the maximum value for each objective amongst both algorithms' results. HV values are found as 0.84295 and 0.84633 for JE_DEMO and JDEMO, respectively. For the JE_DEMO, the objective values are discovered between 4.96 and 6.55 for PDF. From the point of v, range between 0.0057 to 0.0353 is obtained. For the JDEMO, while range between 4.93 to 6.60 is presented for PDF, the alternatives are determined for v between 0.0056 and 0.0328.

From the point of comparing architectural features of JE_DEMO and JDEMO results, the minimum and maximum results of $\mathrm{h_r}$, $\mathrm{d_v}$,$\mathrm{d_h}$, $\mathrm{th_{if}}$, $\mathrm{w_{if}}$, $\mathrm{th_{of}}$, $\mathrm{h_l}$ are similar. However, for $\mathrm{h_{if}}$, $\mathrm{h_{of}}$, $\mathrm{w_{of}}$, JE_DEMO is presented different results than JDEMO. Three alternatives for each algorithm are selected from non-dominated solutions that are shown in Fig.10. We may observe that the results obtained by JE_DEMO and JDEMO algorithms in terms of PDF and v are slightly different from each other. Solutions notated as JE_DEMO3 and JDEMO3, inner frame of height section $(\mathrm{h_i})$ and outer frame of height section $(\mathrm{h_o})$

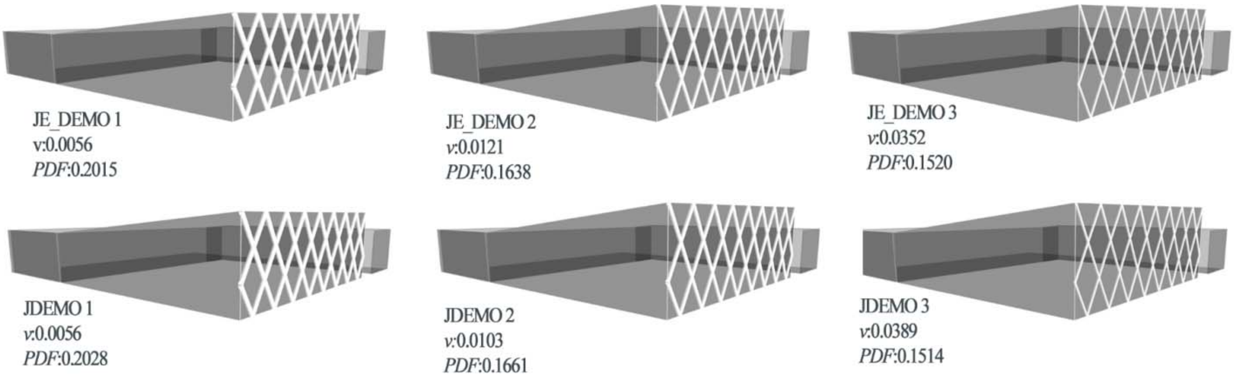

Figure 11 Pareto-Front Solutions attained with JE_DEMO and JDEMO

are less than the other results. Therefore, these results provide more transparency on the diagrid façade from the point of interior space. According to investigation through the non-dominated solutions, it is observed that wider and longer frame elements are presenting better structural performance. However, this fact causes to block natural lighting. To enhance natural daylighting with smaller façade elements, it is noted that size of the thickness of the main diagrid elements are increasing.

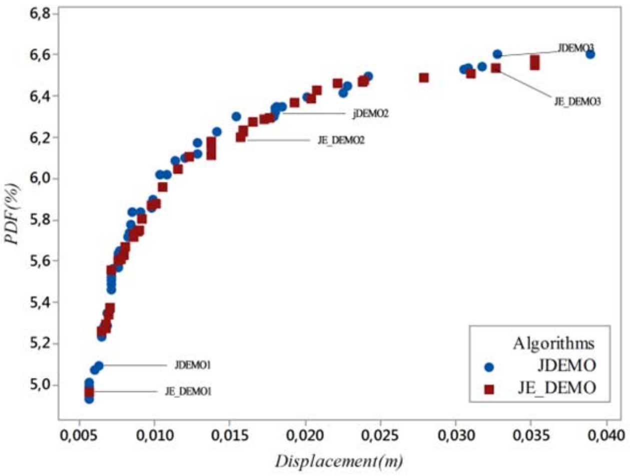

Figure 10 Pareto-front approximations

6. CONCLUSION

In this paper, we addressed an architectural design problem of public pool building using a computational optimization-based approach. For the sake of architectural and engineering concerns, diagrid shape was determined as the main façade element. To reach desirable solutions for this problem, performance of daylight and structure was taken into consideration. For this respect, objective functions were structured as minimization of v that corresponds the structural performance, and maximization of PDF that presents the daylight performance. In addition, to keep the optimization search space in acceptable margins for these objective functions, two constraints are tackled. Two types of the DE, which are JDEMO and JE_DEMO, were compared in order to solve this problem. To present best-trade off alternatives, two algorithms are presented similar objective function ranges at the end. However, as shown in Fig. 10, JDEMO is capable to discover some alternative solutions, which cannot be founded by JE_DEMO, in the search space.

The main contributions of this paper are explained from the perspective of objectives and algorithms. As first contribution, we presented a trade-off between structure and daylight performance. In literature, trade-offs such as daylight and energy or structural performance and material usage mostly studied. However, design of the structure affects the daylight performance, so energy consumption. Therefore, presented trade-off should be carried out during the early stage of the design process, as well. Secondly, we proposed JE_DEMO, which has not implemented to architectural optimization problem before, in this study. To prove the success of JE_DEMO, we compared with J_DEMO, which was used for many real-world and benchmark problems in engineering domain. We observed that JE_DEMO presented competitive results with J_DEMO. Due to No Free Launch Theorem (Wolpert & Macready, 1997), comparing different heuristic optimization methods is highly important. The reason of this is global optimization approach is not possible, which is capable of solving all the problems in the universe. Since design problems are unique because of the location, design concerns, design parameters, user preferences, economical reasons, and plot area, architects/designers/engineers should consider more than one heuristic optimization method to cope with the problem on hand.

For further study, displacement on the glass frame can be considered for different types of façade panels. In addition, façade frames can be evaluated by considering more than one material type to select the most appropriate one. Finally, other MOEAs and swarm intelligence algorithms with different constraint handling methods can be implemented.

REFERENCES

[1] Abbass, H. A. (2002). *The self-adaptive pareto differential evolution algorithm.* Paper presented at the Evolutionary Computation, 2002. CEC'02. Proceedings of the 2002 Congress on.

[2] Abbass, H. A., Sarker, R., & Newton, C. (2001). *PDE: a Pareto-frontier differential evolution approach for multi-objective optimization problems.* Paper presented at the Evolutionary Computation, 2001. Proceedings of the 2001 Congress on.

[3] Bader, J., & Zitzler, E. (2011). HypE: An algorithm for fast hypervolume-based many-objective optimization. *Evolutionary computation, 19*(1), 45-76.

[4] Brest, J. (2009). *Constrained real-parameter optimization with ε-self-adaptive differential evolution*: Springer.

[5] Brest, J., Greiner, S., Boskovic, B., Mernik, M., & Zumer, V. (2006). Self-adapting control parameters in differential evolution: a comparative study on numerical benchmark problems. *IEEE transactions on evolutionary computation, 10*(6), 646-657.

[6] Chatzikonstantinou, I., Ekici, B., Sarıyıldız, İ. S., & Koyunbaba, B. K. (2015). *Multi-objective diagrid façade optimization using differential evolution.* Paper presented at the Evolutionary Computation (CEC), 2015 IEEE Congress on.

[7] Cook, N. (2007). *Designers' Guide to EN 1991-1-4 Eurocode 1: Actions on structures, general actions part 1–4. Wind actions*: Thomas Telford Publishing.

[8] Das, S., Mullick, S. S., & Suganthan, P. (2016). Recent advances in differential evolution–An updated survey. *Swarm and Evolutionary Computation, 27*, 1-30.

[9] Das, S., & Suganthan, P. N. (2011). Differential evolution: a survey of the state-of-the-art. *IEEE transactions on evolutionary computation, 15*(1), 4-31.

[10] Deb, K., Pratap, A., Agarwal, S., & Meyarivan, T. (2002). A fast and elitist multiobjective genetic algorithm: NSGA-II. *IEEE transactions on evolutionary computation, 6*(2), 182-197.

[11] Hernández-Díaz, A. G., Santana-Quintero, L. V., Coello Coello, C., Caballero, R., & Molina, J. (2006). *A new proposal for multi-objective optimization using differential evolution and rough sets theory.* Paper presented at the Proceedings of the 8th annual conference on Genetic and evolutionary computation.

[12] Jakubiec, J. A., & Reinhart, C. F. (2011). *DIVA 2.0: Integrating daylight and thermal simulations using Rhinoceros 3D, Daysim and EnergyPlus.* Paper presented at the Proceedings of building simulation.

[13] Kim, J. (2012). *A methodology for daylight optimisation of facades: An investigation of the opening design strategy with cellular automata for an office building.* Paper presented at the Emerging Technologies for a Smarter World (CEWIT), 2012 9th International Conference & Expo on.

[14] Leonard, J. (2007). *Investigation of shear lag effect in high-rise buildings with diagrid system.* Massachusetts Institute of Technology.

[15] Li, D. H., & Tsang, E. K. (2008). An analysis of daylighting performance for office buildings in Hong Kong. *Building and Environment, 43*(9), 1446-1458.

[16] Madavan, N. K., & Biegel, B. A. (2002). Multiobjective optimization using a Pareto differential evolution approach.

[17] Mallipeddi, R., Suganthan, P. N., Pan, Q.-K., & Tasgetiren, M. F. (2011). Differential evolution algorithm with ensemble of parameters and mutation strategies. *Applied Soft Computing, 11*(2), 1679-1696.

[18] McNeel, R. (2013). Grasshopper 3D.

[19] Mezura-Montes, E., Miranda-Varela, M. E., & del Carmen Gómez-Ramón, R. (2010). Differential evolution in constrained numerical optimization: an empirical study. *Information Sciences, 180*(22), 4223-4262.

[20] Moon, P., & Spencer, D. E. (1942). Illumination from a non-uniform sky. *Illuminating Engineering, 37*(10), 707-726.

[21] Panchal, N. B., & Patel, V. R. (2014). Diagrid structural system: Strategies to reduce lateral forces on high-rise buildings. *International Journal of Research in Engineering and Technology, 3*(03), 374-378.

[22] Perera, A., & Sirimanna, M. (2014). *A novel simulation based evolutionary algorithm to optimize building envelope for energy efficient buildings.* Paper presented at the Information and Automation for Sustainability (ICIAfS), 2014 7th International Conference on.

[23] Preisinger, C., Hemmerling, M., Lemberski, D., Sotomayor, M., Velasco, R., & Günther, P. (2018). Karamba *Informed Architecture* (pp. 157-162): Springer.

[24] Price, K., Storn, R., & Lampinen, J. (2005). Differential Evolution-A Practical Approach to Global Optimization,‖ Springer-Verlag: Berlin.

[25] Robič, T., & Filipič, B. (2005). *DEMO: Differential evolution for multiobjective optimization.* Paper presented at the International Conference on Evolutionary Multi-Criterion Optimization.

[26] Santana-Quintero, L. V., Hernández-Díaz, A. G., Molina, J., Coello, C. A. C., & Caballero, R. (2010). DEMORS: A hybrid multi-objective optimization algorithm using differential evolution and rough set theory for constrained problems. *Computers & Operations Research, 37*(3), 470-480.

[27] Storn, R., & Price, K. (1995). *Differential evolution-a simple and efficient adaptive scheme for global optimization over continuous spaces* (Vol. 3): ICSI Berkeley.

[28] Tasgetiren, M. F., Suganthan, P. N., Pan, Q.-K., Mallipeddi, R., & Sarman, S. (2010). *An ensemble of differential evolution algorithms for constrained function optimization.* Paper presented at the IEEE congress on evolutionary computation.

[29] Tušar, T., & Filipič, B. (2007). *Differential evolution versus genetic algorithms in multiobjective optimization.* Paper presented at the International Conference on Evolutionary Multi-Criterion Optimization.

[30] Waldram, P. (1925). The natural and artificial lighting of buildings. *The Journal of the Royal Institute of British Architects, 32*(13), 405-426.

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

[32] Xue, F., Sanderson, A. C., & Graves, R. J. (2003). *Pareto-based multi-objective differential evolution.* Paper presented at the Evolutionary Computation, 2003. CEC'03. The 2003 Congress on.

[33] Zamuda, A., Brest, J., Boskovic, B., & Zumer, V. (2007). *Differential evolution for multiobjective optimization with self adaptation.* Paper presented at the IEEE congress on evolutionary computation.