Multi-Factor Optimization Method through Machine Learning in Building Envelope Design: Focusing on Perforated Metal Façade

Jinwooung Kim, Jae-Hwan Jung, Seong-Jun Kim, Sung-Ah Kim

Open Science Index, Architectural and Environmental Engineering Vol:11, No:11, 2017 publications.waset.org/10008500.pdf

***Abstract*—**Because the building envelope has a significant impact on the operation and maintenance stage of the building, designing the facade considering the performance can improve the performance of the building and lower the maintenance cost of the building. In general, however, optimizing two or more performance factors confronts the limits of time and computational tools. The optimization phase typically repeats infinitely until a series of processes that generate alternatives and analyze the generated alternatives achieve the desired performance. In particular, as complex geometry or precision increases, computational resources and time are prohibitive to find the required performance, so an optimization methodology is needed to deal with this. Instead of directly analyzing all the alternatives in the optimization process, applying experimental techniques (heuristic method) learned through experimentation and experience can reduce resource waste. This study proposes and verifies a method to optimize the double envelope of a building composed of a perforated panel using machine learning to the design geometry and quantitative performance. The proposed method is to achieve the required performance with fewer resources by supplementing the existing method which cannot calculate the complex shape of the perforated panel.

***Keywords*—**Building envelope, machine learning, perforated metal, multi-factor optimization, façade.

I. Introduction

A. Research Background

SINCE the building envelope is a critical factor in the operation and maintenance of the building, a reasonable design of the building envelope has a significant effect on enhancing building performance and lowering maintenance costs. The aesthetic form of the building envelope is as important as the quantitative performance because it can influence the first impression of the building and even the role of the city's Landmark. For this reason, many architects and engineers have been trying to design reasonable alternatives that simultaneously take into consideration various environmental factors and aesthetic forms when designing the building envelope. The double skin envelope that emerged in the 20th century was one of the best alternatives for achieving qualitative and quantitative performance [1].

Jinwooung Kim and Jae-Hwan Jung are Ph.D. students, and Seong-Jun, Kim is a student with the Department of Convergence Eng. for Future City, Sungkyunkwan University, Suwon, South Korea (e-mail: east6260@skku.edu, jaehwanj@skku.edu, diorkiss@skku.edu).

Sung-Ah, Kim is a professor of the Department of Architecture, Sungkyunkwan University, South Korea (corresponding author, e-mail: sakim@ skku.edu).

Fig. 1 Example of installing a perforated panel with pictures (a) Chicago Arkadia Apartment (b) Nordstjerneskolen state school (c) Tokyo Dior store

In modern buildings, the exterior of the building has become more attractive because the shape of the building and the exterior of the building are separated, and an abstract dimension is given throughout the building. Especially, the double skin made of perforated panels can control the quantitative performance through incident solar radiation, sound insulation, air flow, heating and cooling load, and openings. Representative examples of these attempts are the appearance of exterior designs that emphasize design elements by expressing desired images as patterns of perforation. As the number of building envelopes composed of perforated patterns increases, many design methods have been attempted to improve the quantitative performance, reflecting the images rather than the patterns to implement them (Fig. 1). However, to optimize the various performance of the panel at the same time, we face the problem of time and the limit of calculation tools. Some of the current optimization processes Fig. 2 uses an infinite number of alternatives to detect alternatives that designers have not considered during the design phase, optimize them using arbitrary variables and without proper reasoning [2]. In other words, the general optimization steps consist of these three steps. 1) Create alternatives by randomly reflecting variables within the designer's specified range of variables. 2) Analyze design alternatives through the BPS (Building Performance Simulation) tool. 3) Repeat the above procedure until you get the best solution [3]. This optimization process differs in the amount of time and resources the process takes, depending on the amount of time spent in first and second, especially in evaluating alternatives using the two BPS tools. Overwhelming BPS tools have the advantage of helping architects make better decisions when designing by providing

Open Science Index, Architectural and Environmental Engineering Vol:11, No:11, 2017 publications.waset.org/10008500.pdf

performance results. However, performance analysis through BPS is limited in the optimization process because it takes much time to support the alternative search in the initial design stage efficiently [4].

When designing a double skin envelope using a perforated panel, consideration should give to the size of the perforation, placement, identify with the image, and quantitative performance. The BPS tool requires resources that are proportional to accuracy or complexity to increase the accuracy of the analysis or if the shape of the analysis target is complicated (the amount of data about the form is significant). Perforated panels shape is too complicated and relatively small in size compared to buildings, so they have high accuracy and complexity. To optimize this envelope, therefore, new methods are needed, not optimizing over time and computing resources.

If machine learning that supports heuristic search applied to the design stage, which suggests the possibility to utilize BPS efficiently. GA (Genetic Algorithm) is a representative example. It is a methodology based on evolutionary experience and reasoning that better offspring will reach their highest goal. By using machine learning, we can optimize for optimal alternatives based on experimental and empirical reasoning rather than searching countless options. In the simulation phase of the BPS, if the simulation provided by inferring various situations based on practical experience, the time required for the optimization process will reduce, which can enable immediate feedback on the alternative. The DNN (Deep Neural Network) algorithm can support BPS that supports performance optimization at the initial design stage because heuristic through much experimental data is possible, and ultimately can reduce the time required for optimization.

B. Research Aim and Methodology

This study proposes and verifies a method to optimize the dual envelope of a building composed of perforated panels by using machine learning following design shape and quantitative performance. Rhino Ceros and Grasshopper utilized for this, purpose and DNN algorithm was implemented through Rapidminer™ As a BPS, the insolation analysis carried out with the Grasshopper Add-on Ladybug. The experiment consists of four stages. In the final step of verification, the proposed methodology, the method using GA, and the optimization method using general BPS tools compared and verified.

II. Theoretical Background

A. Optimization Methodology and Genetic Algorithm

Optimization is the process of finding the minimum or maximum value of a function by selecting multiple variables with different constraints. The optimization function is called cost, suitability, or objective function and is usually calculated using simulation tools. There are genetic algorithms as examples. Particle Swarm Optimization (PSO), which is a particle cluster optimization algorithm, and hybrid algorithms, which made by combining various algorithms, are also available [5]. Genetic Algorithm (GA) is merely an optimization process that mimics biological evolution. In nature, organisms have survived to adapt to their environmental context by evolving their genetic makeup. This adaptation process is driven by three fundamental principles: natural selection, recombination of genetic characteristics, and mutation.

A similar primary can utilize in software programming. The GA program uses symbolic steps such as assessment and selection, recombination, and mutation to optimize initial alternatives. First, in the initialization phase, quantities of different individual alternatives are generated. As with nature, it is essential that there are enough populations for successful evolution. The steps are then evaluated based on a set of suitability and selected by the programmed conformance function. The more likely an alternative is, the more likely it is that the alternative is programmed to "survive" and the right alternative chosen. The next step is to spread the population through the recombination of genetic characteristics. Recombination constitutes a new individual from the genome of a viable solution. The mutation function also exists to allow population diversification and variant formation. Mutations prevent optimal regional occurrences and ensure diversity in the next cycle. These three steps repeated, and each cycle represents one step in evolution. The process repeats itself periodically and continuously until the desired breakpoint reached, or until all new variants in the system are less suitable than the existing population [6].

In some cases, a new algorithm developed by modifying genetic algorithm among hybrid algorithms. Hamdy [7] initially created a new combination of algorithms that prevents GA's random analysis and iterates until a high-quality alternative arrives at the goal. There is also a study by Hasan [8] that optimizes the cost of the building by lifecycle using hybrid algorithm consisted in the PSO, which is more efficient to find global optimization and Hooke-Jeeves algorithm, which is more efficient to find local optimization.

There have been many studies to optimize the maximum and minimum values in the optimization process. In the same vein, there are many heuristic optimization methods, but most of them used for optimizing finding the numerical minimum or maximum value rather than using the heuristic in the optimization simulation stage.

B. Machine Learning and Deep Neural Network

A machine learning system is used to identify objects in an image, copy voice to text, match news items, posts, or products to a user's interest, and select relevant results of the search. Gradually, these applications use a technology class called Deep Learning [9]. The key to deep learning is that humans do not design the middle layer that has the function. It learns from the data on its own using general-purpose learning procedures. That is, the heuristic is possible through learning the algorithm itself, and the reliability of the heuristic is also proportional to the amount of data.

C. Heuristic Optimization Method

1) Preliminary Study

The term 'heuristic optimization' is a widely used term and is an empirical way to find optimization directions in [10], also used the term 'Heuristic optimization,' but modified the GA for 'Global Minimum Finding' to find the optimal solution. The 'Hybrid Optimization' proposed by GenOpt. [11] Also, deals with heuristic optimization issues. Gen Optics uses a heuristic-based PSO to find an alternative Global Minimum Domain, and after finding the global minimum, Hooke-Jeeves method is used to determine the optimal value. Most 'heuristic optimization' studies focus on the search for optimization issues, i.e., the Global Minimum.

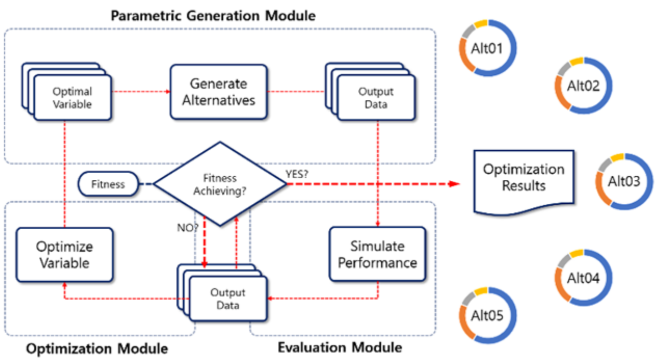

Fig. 2 Simulation-based optimization process [12]

As shown in Fig. 2, the simulation-based optimization process takes several steps to return one optimization cycle. The process suggests alternatives by setting the performance and optimization algorithms what requested achieve initially. That is, from the GA point of view, the process of choosing the best alternative (viable alternatives) by propagating new genetic traits occurs in the "Optimization Program." A viable alternative, that is, an alternative approach to the target performance, is checked to see if it achieves the target performance. If it does not achieve the target performance, it is a recursive process that goes through the step of "Evaluate Performance" again.

In the "Optimization Program," the process of assessing whether or not a survivor is high, i.e., the degree to which quantitative performance has achieved, is analyzed in real-time in the "Evaluate Performance" and reflected in the loop. However, the BPS-based optimization process has a limitation in that there are many resources consumed in measuring the "Evaluate Performance." For example, assuming that "Evaluate Performance" takes 30 seconds and the population according to the variable is 10,000, the time it takes for a generation to cycle is about 3.5 days. The total sum of Computational resources across generations is enormous, even if the resource requirements for "Evaluate Performance" are small.

The heuristic used in the previous study mentioned above is not to evaluate 10,000 times analysis but to use additional resources by proceeding from the alternative which is assumed to be the most probable by using the heuristic. The 'Simulation-based optimization' technique is a method of lowering the resolution of a simulation or saving a resource by extracting a part of a building, creating a digital prototype of a similar type, and applying a limited simulation.

2) Concept of Heuristic BPS

Heuristic BPS proposed in this study uses Heuristic in BPS phase using DNN. 'Heuristic Building Performance Simulation' can apply 'Heuristic' to the simulation stage to analyze the performance of the building during the optimization process, thus saving the resources consumed for optimization. In Fig. 3, the first or seconds' generation alternatives are derived, and the performance analysis is carried out through the data obtained from the Input File.

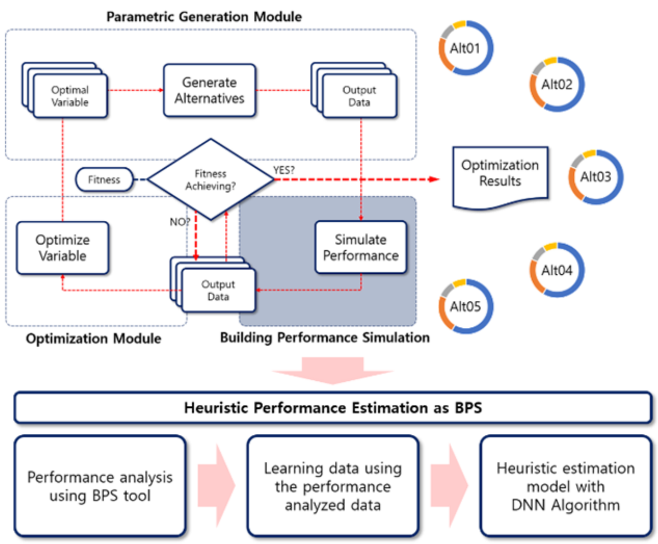

Fig. 3 The concept of Heuristic optimization process

The performance analysis shows that the variables X_1, X_2, X_3 ... X_n and labels Y_1, Y_2, Y_3. ... Y_m, and the relationship between the input variable and it is summarized as Train Data. When the learned BPS tools analyze the early-generation alternatives, the resources are similar to the traditional way, but the more the generation repeated, the less the resource requirement. Because of heuristics. For example, in a factory, heuristic BPS plays a role of a mold for mass production, so it is quick to make it through a mold compared to producing a product manually. The heuristic BPS takes time to implement the analysis model that plays the role of the mold initially, but as the analysis model becomes more and more developed, the resource required decreases. The Heuristic BPS Based Optimization proposed in this study applied to the heuristic-enhanced BPS optimization process.

III. Methodology

Fig. 4 indicates the overall implementation of 'Heuristic BPS.' The first to third phases are the steps to implement the BPS, and the final step is to derive the control group alternative for verification. The first step is to make the image a panel. When an image input through Grasshopper, it implements an algorithm that automatically creates a perforated panel based

Open Science Index, Architectural and Environmental Engineering Vol:11, No:11, 2017 publications.waset.org/10008500.pdf

Open Science Index, Architectural and Environmental Engineering Vol:11, No:11, 2017 publications.waset.org/10008500.pdf

on the value of a picture pixel.

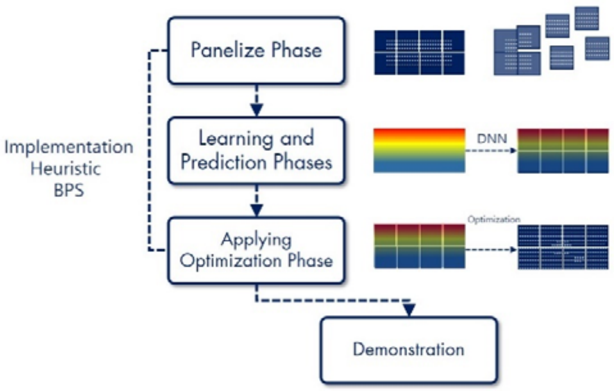

Fig. 4 Implementation heuristic BPS

The second step is to create a DNN learning model. The BTS is used to analyze the shading performance of the panel and the panel, which determined by designating the arrangement of the perforations having different radii and sizes. Using the analyzed data (training), we create sample data and create a DNN learning model and analyze the accuracy. The third step is to apply the DNN to enhanced BTS level to the optimization process to derive an alternative.

A. Panel Generation Algorithm

The paneling algorithm created with Grasshopper. There are several settings for the elements that make up a panel to make it. The size of the panel defined as 1m x 1m in consideration of the production process, and the number of perforations set to 100 per panel. Since the number of perforations per panel affects the resolution of the image, the number of perforations increases the resolution, which is disadvantageous to computation. Since the slit cut in half or one-quarter at the interface, the perforation array is an array of 11x11, but the perforation area is equal to 10x10. The maximum value of the perforation radius defined by dividing the length L of one panel by 10, the number of spaces between the perforations, and not exceeding 95% of the length.

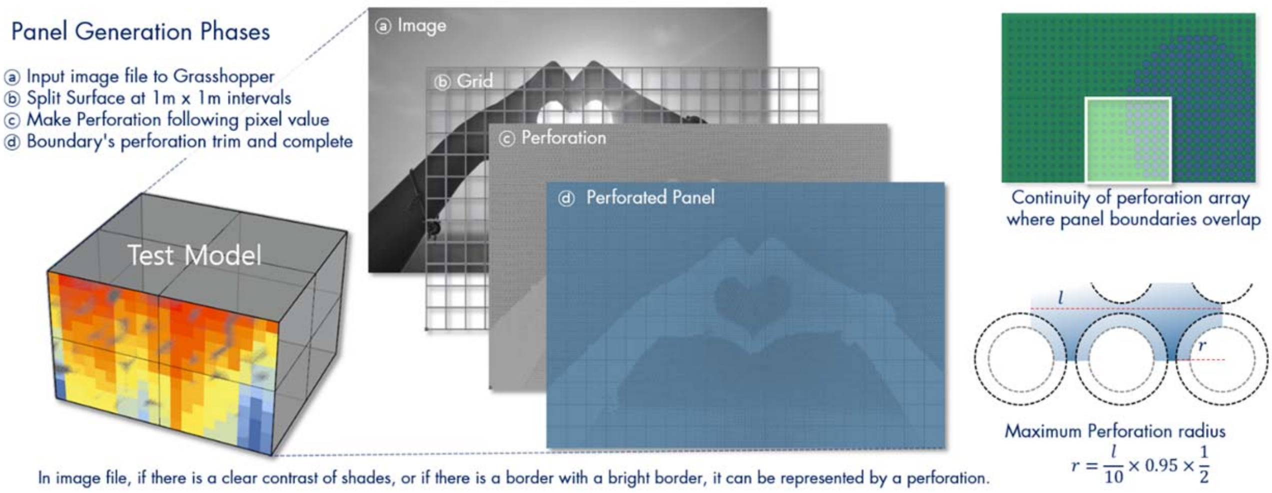

Fig. 5 Panel generation phase. It has four stages of the process. In an image file, if there is a clear contrast of shades, or if there is a border with a bright border, it can be represented by a perforation

As shown in Fig. 5, the panel generation algorithm consists of four steps. In step ⓐ, the grayscale value of Image extracted from Grasshopper. After splitting the facade area to be applied in step Ⓑ to 1m x 1m, in step ⓒ, create the perforation using the grayscale value derived from the step ⓐ. At this time, the maximum value of the perforation radius is the same.

Since the maximum value of the grayscale value is the maximum value of 'r,' and the minimum value is equal to the minimum value of the grayscale value, the 'r' depends on the grayscale value and is proportional. In step ⓓ, if the perforation is on the boundary, set it to include in each panel on both sides. A panel can be created by combining the non-perforated surfaces to create the surface and then setting the thickness. The building selected for the experiment is a building located in Myeong-Dong area, Seoul. It is a south-facing building with a height of 12m and a width of 20m. Therefore, the total number of panels is 240, and 24,000 perforations can be formed.

B. Heuristic BPS Optimization

Grasshopper add-on program, Ladybug Radiation analysis, is used to derive the average solar radiation per year. The target time for the average solar radiation is 12 hours from 8:00 am to 8:00 pm and the analysis area set to the average sunshine per 1m2 of the size of one panel. For maintaining the image shape in the optimization process, the perforation size was maintained by classified. According to the algorithm of the previous step that created the perforation according to the grayscale value, there are 24,000 perforation types in the whole panel. If specified the radius of these perforations as a variable, must reduce the number of variables because the number of variables is too large. Therefore, the distribution of grayscale value is

Open Science Index, Architectural and Environmental Engineering Vol:11, No:11, 2017 publications.waset.org/10008500.pdf

divided into ten and classified into ten types according to perforation size. We also set control factors for each of the ten types of variables so that the types do not change during the optimization process. Because, if the type changes, the shape of the image may change after the optimization process, the type must maintain. Size divides ten types, and the range of each type not duplicated.

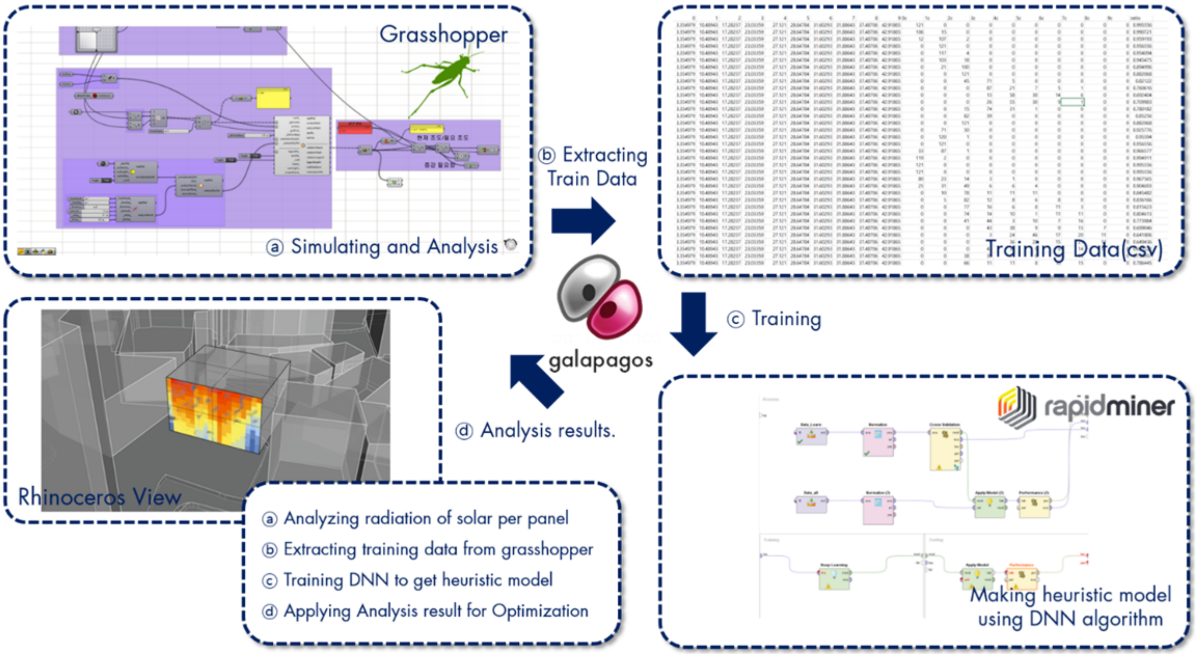

Fig. 6 Concept diagram Heuristic BPS. As a recursive process, a series of processes that extract data from GH, train through data and apply trained heuristics to optimization

Fig. 6 analyzes the amount of sunshine without applying the panel in step a. The output of step a can be visualized in Rhinoceros and can also be stored in the form of data. The data stored in units of the kilowatt-hour (kW / h) of sunshine for each panel. In each part, it derives an over or under 200 kW / h of required illumination. In step b, the material DNN can learn. This data, called Train data, consists of a variable and its LABEL. A total of 20 variables entered in the variable, and one label follows.

The variables are the number of ten types of perforations applied to the panel and the radii length of ten per type perforations. The label defines the shading rate, that is, the rate at which one panel can block light. We implement the learning model in step c using data paired with each variable and corresponding label. For example, when there are 100 perforations in one panel, there are 97 perforations of 0 type and three perforations of 1 type. When the shading rate is 94%, the learning model learns 48 perforations of 0 type; it is also possible to deduce the shading rate when 52 types of perforations are present. Of course, the amount of data must be tremendous for the learning model implement.

The model learned in step c is panelized when the image is input. At this time, it is possible to calculate the light amount without analyzing the ladder of the grasshopper by analyzing the perforation of each panel. Heuristic BPS is possible if performance can infer without simulation tools through learning. In step d, the data analyzed by the heuristic BPS is used again for the optimization process.

IV. Experiments

The following three optimization processes compared and analyzed based on the degree of image implementation and the extent of achievement of the desired performance.

(A) Computation(BPS) + Galapagos
(B) Ladybug (BPS) + Galapagos
(C) Ladybug (Heuristic BPS) + Galapagos

Since the time spent 's hard to implement the performance of the computer or other control conditions, it should note that this is not an objective indicator but is a reference factor.

A. Computation (BPS) + Galapagos

In the first case (A), instead of the 'Ladybug,' the amount of light lost was calculated through the ratio of the area of the perforation to the area of the panel. Therefore, the amount of sunshine due to scattering and reflection of light not calculated so that that difference may occur. That is, it assumed that the BPS performance lowered because, in the BPS part, it does not use BPS tool such as 'Ladybug,' but acts as a BPS that only consisted computation.

The amount required to reach the initial target illuminance of 200 kW / h, which is the required performance, is defined as the 'shading requirement.' That is, assuming that the amount of sunshine of the current panel position is 400 kW / h, only a light amount of 50% is required. Therefore, assuming that the light penetrates 100% of the perforated portion, the percentage of the perforated part of the panel area of 400 kW / h should be 50%.

Open Science Index, Architectural and Environmental Engineering Vol:11, No:11, 2017 publications.waset.org/10008500.pdf

$$\text{Shading requirement, N} = \frac{Required\ \ Radiation\,value}{Panel\,Radiation\,Value}$$

$$\text{Shading ratio, B} = 1 - \frac{Perforated\ Area}{Panel\ Area}$$

The optimization process proceeds to the Galapagos and the value of 'r' for each type of perforation radius type that the value of 'B' for each panel can best match 'N.'

When the 'r' value derived by the Galapagos was substituted, the target 200 kW / h achieved by 78% of the panel. The lowest panel is 54.98%, and the highest panel is 99.97%. In the case of 22% which does not satisfy 200 kW / h, it analyzed that the arrangement of perforation size is fixed to maintain shape. The performance varies according to the constraints for the R-value range, and in particular, 100% of the panels achieved by eliminating the placement constraints of the perforation (as shown in Fig. 7). However, since the shape has become indistinguishable, performance weights must be adjusted between image shape re-implementation and delivering sunshine. The operating time of the Galapagos algorithm took more than 8 hours.

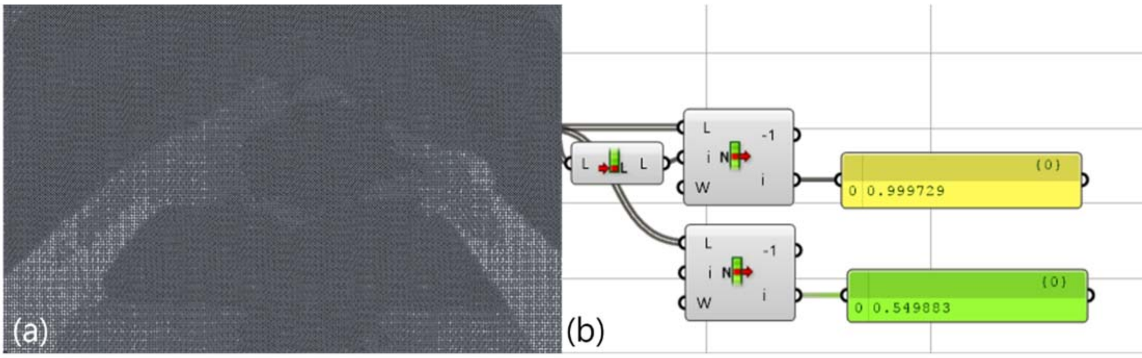

Fig. 7 (a) Eliminating the placement constraints (b) The lowest and the highest achievement of required radiation

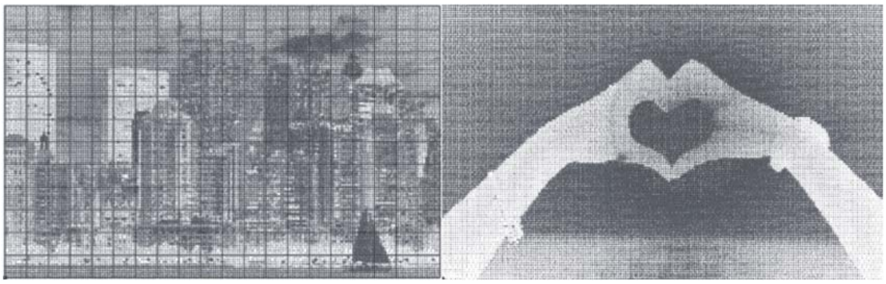

Fig. 8 Generate perforated panel using heuristic BPS

B. Ladybug (BPS) + Galapagos

In the second case, the 'Ladybug' kept at the level of the level used in the performance analysis of the Façade where the original panel's located, and heuristic BPS was not used. (B), the amount of sunshine due to light scattering, reflection, refraction is reflected, so that the accuracy of BPS is high. The 'Ladybug' as a BPS has the highest efficiency, but the longest computation time, and the maximum resource spending among the methods. The value of 'r' for each perforation radius type is derived from the value of the light shielding rate B, and the light shielding required amount N as in the case of (A). The difference is that the light shielding rate B is derived by 'Ladybug, ' and the accuracy is improved.

The optimization process proceeded to the Galapagos. However, it took 3.7 minutes to analyze one alternative in 'Ladybug' as BPS, so we could not measure the time to compute one generation in Galapagos optimization. Even, one generation could not be analyzed because one generation consisted of more than 1000 alternatives on average. However, if it set the resolution lower enough to compute quickly in the 'Ladybug,' there was no reason to simulate performance. If the value of resolution becomes as small as calculate quickly in 'Ladybug,' the resolution becomes larger than the panel size. So, it is impossible to simulate the shading through the perforation because the resolution is equal or greater than the panel size.

C. Ladybug (Heuristic BPS) + Galapagos

This third method, Heuristic BPS, required the most resources to create training data. Although generating training data is also assumed to be the work scope of Heuristic BPS Optimization, there is a difference in efficiency compared to (B). 'R' value for each type of perforation radius type is derived from the value of the shading ratio B and the shading necessary amount N in the same manner as in (A). Galapagos proceeded as above, but the difference is Heuristic BPS applied in optimization.

The layer for implementing DNN was composed of 250 neurons in five layers. The accuracy of the heuristic BPS is 99.13%. In Fig. 8, the graph on the left is the learning data, and the chart on the right is the graph of the amount of light deduced empirically. Train Data has a total of 21,121 variables. In the case of the third heuristic BPS, the degree of achievement of minutiae is analyzed to be around 79%. As in the first instance, the limit of optimizing the shape of the image was examined to be in the latter half of 70%.

Although the accuracy of the heuristic BPS is 99.13%, there is a physical limit to the optimization of minutiae. However, in the case of heuristic BPS, it was possible to analyze the training data in less than 3 minutes and the learning time in less than 3 minutes. After learning completed, the real-time analysis was possible.

V. Conclusion

In the process of optimizing the building envelope composed of perforated panels, we propose an optimization method that can overcome the complexity of the shape due to the perforation and the limitation of the simulation tool. Time efficiency or accuracy is superior to simulation-based methods, especially time efficiency.

The empirical BPS was found to be more suitable for the optimization process than the simulated BPS. Simulation-type BPS requires data to be obtained once the operation of one operation completed, and data once used in the optimization process not reused. However, the empirical BPS learns the data derived from the past computation in the optimization process, so the reliability of the process increases as the process progresses. Obtaining good quality training data is hard. Therefore, research on the process of processing high-quality data and data should continue.

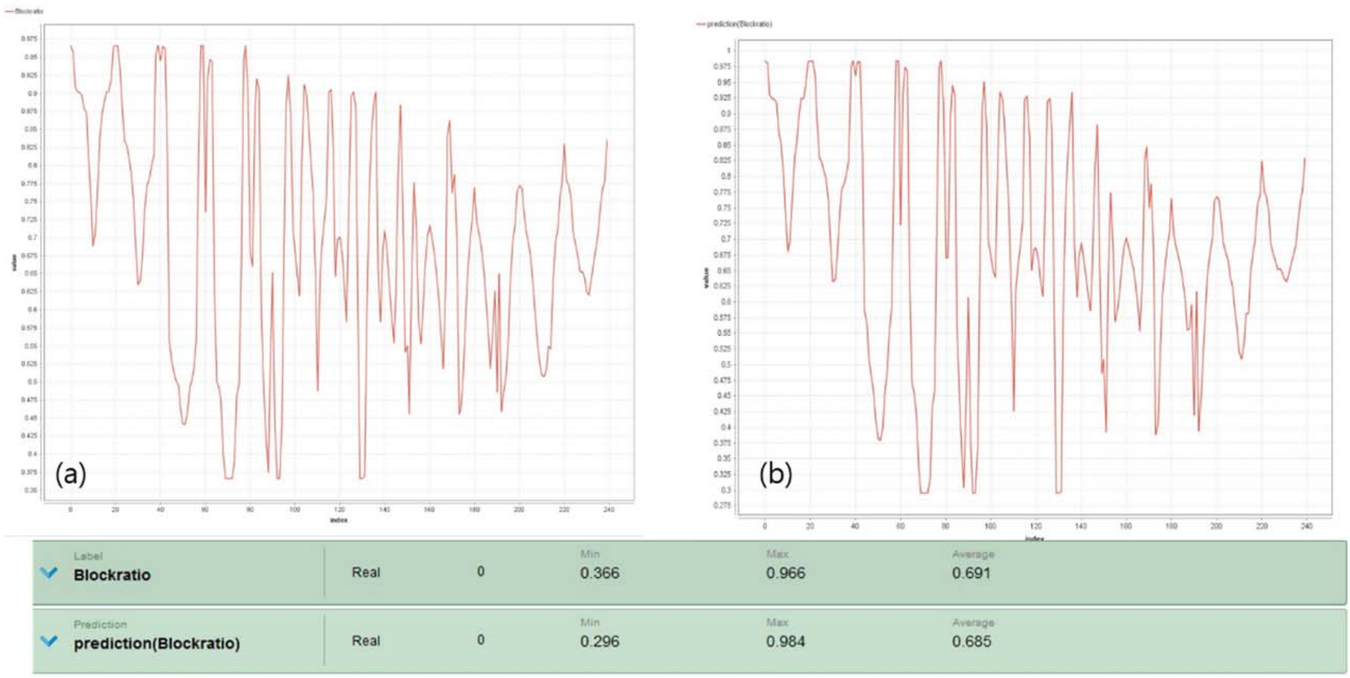

Fig. 9 (a) Training Data graph, (b) Prediction Data graph

The inference is based on data and is the number of cases that occur with the highest probability for the future. Simulation is the same. A simulation is a proof derived from an operation, but the formula or algorithm that it computes is not the actual thing, but the most likely thing is to check the phenomenon virtually. In other words, since the variables for all situations cannot be considered, the computation through simulation is only possible with high probability. On the contrary, if high-quality data can be accessed, empirical inference and fast feedback are confirmed to have little difference in reliability compared to simulations that rely on simple computation.

Acknowledgment

This work is financially supported by Korea Ministry of Land, Infrastructure, and Transport (MOLIT) as 'U-City Master and Doctor Course Grant Program' and supported by Basic Science Research Program through the National Research Foundation of Korea (NRF) funded by the Ministry of Education (NRF-2016R1D1 A1B01013694).

References

[1] Blanco, J. M., Buruaga, A., Rojí, E., Cuadrado, J., & Pelaz, B. (2016). Energy assessment and optimization of perforated metal sheet double skin façades through Design-Builder; A case study in Spain. Energy and Buildings, 111, 326-336.
[2] Deutsch, R. (2017). Convergence: The Redesign of Design. John Wiley & Sons.
[3] Nguyen, A. T., Reiter, S., & Rigo, P. (2014). A review on simulation-based optimization methods applied to building performance analysis. Applied Energy, 113, 1043-1058.
[4] Yi, Y. K., & Malkawi, A. M. (2009). Optimizing building form for energy performance based on hierarchical geometry relation. Automation in Construction, 18(6), 825-833.
[5] Optimization of Constrained Function Using Genetic Algorithm. (n.d.). Retrieved from http://www.iiste.org/Journals/index.php/CEIS/article/viewFile/35839/36834 Accessed on 02/10/2017
[6] (Machairas, V., Tsangrassoulis, A., & Axarli, K. (2014). Algorithms for optimization of building design: A review. Renewable and Sustainable Energy Reviews, 31, 101-112.)

Open Science Index, Architectural and Environmental Engineering Vol:11, No:11, 2017 publications.waset.org/10008500.pdf

[7] Hamdy, M., Hasan, A., & Siren, K. (2011). Applying a multi-objective optimization approach for design of low-emission cost-effective dwellings. Building and environment, 46(1), 109-123.

[8] Hasan, A., Vuolle, M., & Sirén, K. (2008). Minimisation of life cycle cost of a detached house using combined simulation and optimisation. Building and Environment, 43(12), 2022-2034.

[9] LeCun, Y., Bengio, Y., & Hinton, G. (2015). Deep learning. Nature, 521(7553), 436-444.

[10] Rüdenauer, K., & Dohmen, P. (2007). Heuristic Methods in Architectural Design Optimization

[11] Wetter, M. (2009). Generic Optimization Program User Manual Version 3.0. 0. *Lawrence Berkeley National Laboratory*.

[12] Kim, S. W. (2009). A Classification and Optimization Method of Atypical Building Panels by Panel Production Methods. Yonsei University, Dept. of Architectural Engineering, Master's Thesis

Open Science Index, Architectural and Environmental Engineering Vol:11, No:11, 2017 publications.waset.org/10008500.pdf