A GENERATIVE TOOL BASE ON MULTI-AGENT SYSTEM

Subtitle: Algorithm of "HighFAR" and Its Computer Programming

LI BIAO, LI RONG, XUE KAI, LIU CHANG AND GAO QIN
School of Architecture, Southeast University, Nanjing, P.R. China
jz_studio@126.com, lr19710114@126.com, ovni_xk@hotmail.com, ovni_xk@hotmail.com, nutsgq@163.com

Abstract. Utilizing the principle of multi-agent system by computer programming, the paper presents achievement of an architectural generative design tool which implements intelligent architecture design with the focus of the layout in Asian high density "Floor Area Ratio" (FAR). Applying with the software, architects can get a high density FAR planning design in several minutes. From macroscopic position of building agents in geography to encode the agents by computer programming, the paper expounds the process of the generative tool and its mathematics algorithms.

Keywords. Multi-agent system, FAR, genetic algorithm, generator.

1. Background

The definitions of multi-agent system (MAS) in different fields have different concepts, for example, agents in nature are matters of surviving and developing in competition so that controversies occurred. Sometimes, multi-agent system is claimed to be self-organized system or autonomous activity. In computer science, a multi-agent system is a system composed of agents showing collective ability of reaching goals that are difficult to achieve by an individual one, and every agent has its duty to find resources and to match the resources of the whole system without bidding. Actually, it is difficult to research an ideal effect of completely independent in each field; all agents are under controlled by human supervision. Moreover, interdependent systems are needed instead of completely autonomous ones. However, employing the principle of multi-agent system, many tools hinting theory and the practical value can be developed by computer algorithm.

As an attempt at new method for architectural design, in spring semester of 2007, eleven students, who knew little about computer programming originally, took part in our group of generative design. They studied principle of generative design by computer programming, such as the case of "game of life" and others, which were based on the principle of "Multi-Agent System". Meanwhile, by the development of computer games and small generative exercises, they implemented the transition from ignorance of generative principle or methods to theoretically comprehend and ability of computer programming groups. For their final works, the students were divided into 4 groups. Whatever it is difficult to get many achievements of generative architecture design tool within only 4 months, but benefiting from an accurate schedule, the students implemented four architectural generative tools, which came from the four groups. The case of this paper achieved intelligent architectural design software with the focus of the layout of Asian high "Floor Area Ratio" (FAR) demonstrated capacity of the principle employed multi-agent system.

2. Fundamentals of High FAR

Actually, from a certain point of view, the planning layout of high-rise residential buildings has the characteristics of general multi-agents system; coexistence of all residential buildings maps the fundamental characteristic of MAS operation. Furthermore, each residential building as one independent agent is related and restrained with others, which turns the entire planning layout of high-rise buildings into the manifestation of autonomy and activity. The characteristic of agents' coexistence, which shows complex nonlinearity relationship between each others, makes the process of plan layout to be considered as a complex adaptive system.

Employing the model of MAS and simple genetic algorithm, "HighFAR" integrates multi-possibilities and generates high-level residential district planning and demonstrates a huge superiority. Meanwhile, the generative tool offers an interface for input parameters, such as range and footprint area of buildings height, and etc, which ensure generative achievements colorful (see *fig.1*: the interface of "highFAR" and its input parameters).

Figure 1: Interface of "High-FAR"

2.1 RULES OF HIGHFAR

From the view of architectural design, all the buildings must keep away from a defined distance (for fire protection, sunlight requirement or others), and the distance should be set within a rational distance in order to get state of high FAR. The tool of "HighFAR" employed kinds of attractive and repulsive forces between every two agents until get a global balance. The main rules are presented as follows:

Rule1: Each two agents compare distance with a defined parameter of distance, such as 13 meters for fire protection. If the distance is less than the defined parameter (13 meter), the two agents will shift toward opposite directions along the line of the two agents.

Rule2: Every agent of building has its sun shadow, but the shapes of the shadow vary with time variation and form of the building. The generative tool provides interface for customer to input the parameters, such as a scope of time for generating shadow of building. By the parameter of time range, all agents can drop their shadows on the ground according to shape and height of the buildings, and hence the shape of the shadows becomes kinds of "forbidden zone" for other agents. Each agent can't be located in the "forbidden zone", otherwise, it can't get enough sunshine (*Figure. 2*).

Figure 2: Keep away from the "forbidden zone"

Rule3: The number of agents on one defined site must be settled as more as possible, which is a basic condition for generating high FAR planning layout.

Rule4: The range of building height and radius of building, such as 50 meters to 100 meters for building height and 20 meters to 30 meters for plan radius, are two parameters for customer to input. With the parameters, the tool can generate diversified achievements.

2.2 AGENTS ENCODING

Before the start of computer programming, it is one of the most important steps for agents to be encoded and transformed the codes into graphical output. Applying methods and properties of OOP (OOP, Object Oriented Program) from computer programming development, the tool refines the essence of agents shape as circles, and set their position, height, radius as properties of every instance. The class of "Agent" offers many methods for calling from main program during its running time, such as "attract ()", "repel ()", "calculateDistance ()" and etc.

Employing the knowledge of physics, mathematics and the position of building on earth, it is not difficult to figure out shape of shadow region according to shape of buildings. Correlation formulae show as follows:

$$\sin(hs) = \sin\Phi \times \sin\sigma + \cos\Phi \times \cos\sigma \times \cos\Omega\ ; \qquad (1)$$

$$\sin(As) = (\sin(hs) \times \sin\Phi - \sin\sigma)/(\cos(hs) \times \cos\Phi)\ . \qquad (2)$$

Among them: hs is solar altitude;
As is solar azimuth;
Φ is latitude;
σ is longitude;
Ω is hour angle.

$$\sin(hs) = \frac{-\sin\sigma\sin\varphi + \sqrt{\sin^2\sigma\sin^2\varphi - (1-\sin^2(As)\sin^2\varphi)(\sin^2(As)\cos^2\varphi - \cos^2\varphi\cos^2\sigma)}}{1-\sin^2(As)\cos^2\varphi}$$

According to the above two formulae, another important formula could be deduced, the complex deduce steps omitted here, and the result is:

(3)

Utilizing the above formula, the shadow of building involved range of time may be calculated (Figure. 3). Agents encoding concerning Rule1 (Showed in 2.1) is easier for computer to calculate, and the formula is:

$$dist = \sqrt{(x_1 - x_2)^2 + (y_1 - y_2)^2} \qquad (4)$$

Among above: $dist$ is the distance between two agents;
(x_1, y_1) and (x_2, y_2) are positional coordinates of two agents.

Figure 3: Different period of time leads to different shape of "forbidden zone"

2.3 REPELLENT FIELDS AND AGENTS GENERATED

The “domain” of each agent, where other agents can’t enter, contains two parts: one part originated from “*Rule1*” in sections 2.1 is a radius district which keeps other agents away (*Figure. 4_1*); another part is the shadow originated from “*Rule2*” in sections 2.1 is a district of polygon (*Figure. 4_2*), which can be deduced from “formula A” showed above. As the results, the “domain” of agents which standard by “forbidden zone” is the combination of the two parts (*Figure. 4_3*). When one agent is located in others “forbidden zone”, it will move to opposite direction until they get separating state (*Figure. 1*).

Figure 4: Composition of “forbidden zone” or “domain” of agent

It is difficult to figure out how many agents of buildings for a building site can hold. Our tool endows life characteristic with its agents of buildings. Each agent generates at a random position on building site one by one, and detects whether it is in a “domain” of others: If true, then they push each other to opposite direction; on the contrary, it locates at the random position which maybe changed due to the adding of other agents later. By this means, the building site will fill with agents as more as possible till it gets state of saturation, and each agent occupies their own “domain”.

As we all know, it is uncertainly that every agent of building sets its highest height will make a planning layout get a high FAR, many other factors, such as radius of apartment plan, numbers of agents and etc, influencing ultimate outcome should be considered. Different building shape will drop different shadow on building site, and hence restrict the number of agents. Furthermore, the height of building is generated within a scope (such as 80 meters height to 100 meters height) which is a parameter input by customer. The value of the scope ensures diversification of ultimate outcome; at the same time, the generative outcomes could keep high FAR.

An appraisal system will evaluate each outcome generated by the tool. Apart from the floor area ratio must be considered, another parameter named “occlusion rate” which is a highly abstract indicator of the best direction for “HighFAR” to evolve. The generative tool utilizes simple genetic algorithm for its final step: it records the data of the best result as a “record-holder” which will compare it with other results generated later, and the running of “HighFAR” can be terminated by several means, such as the time lasted, rounds it has made or terminates by customers.

Based on the analysis above, a flowchart of the generative tool emerged (see figure 5).

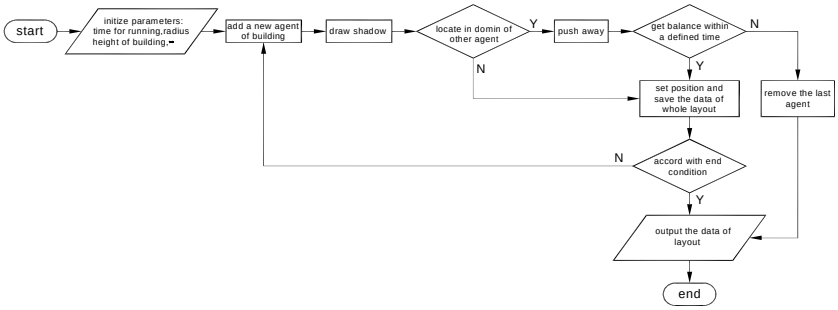

Figure 5: Flowchart of the tool

Another tool, as a generative tool for "skin" of buildings, is accessory software of "HighFAR". Its generative principle base on strong relation between area of windows and its corresponding rooms, such as the ratio of a bedroom is "1/7" (area of window/area of bedroom), ratio of corridor is "1/10" and so on. The skin generator can locate all windows randomly but correspond to the area of rooms behind (see *Figure 6*).

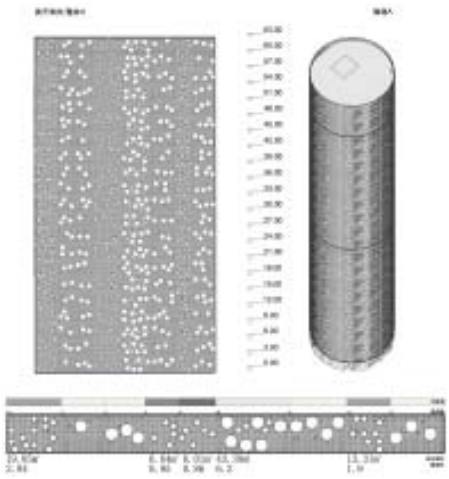

Figure 6: Skin generator

3. Inputs the Real Site

An irregular shape of approximately 66000 square meters building site, which limited the range of all agents' motion, was input as a vector diagram for testing the software after the development had finished. It's a real site in Nanjing, China, and there is a river to the west of the building site, which offers suitable condition for testing "occlusion rate" of direction to evolve.

By the way of traditional method, it needs architects to take a long period of time to arrange building layout. Moreover, they must make the multi-plan comparison for achieving an optimum layout of all regional distribution, which is a tedious and arid work. Applying the high efficient generative tool, it becomes an extremely simple task for alteration the scope of building height, specification of architectural plan and regenerating a new layout, and lots of optimized results can be generated by "HighFAR" within a short period of time (*Figure. 7*).

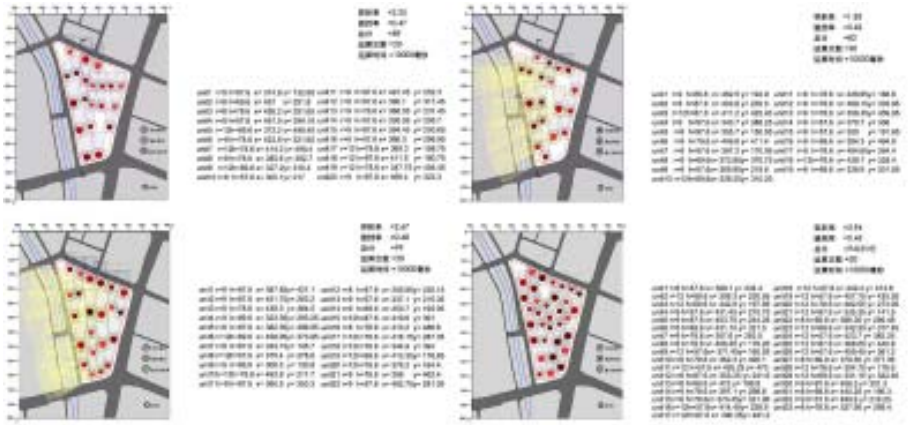

Figure 7: Several results generated by the tool

By analyzing the example, relevant concepts of evaluation for generated results can be elucidated as follows:

- FAR: Floor Area Ratio

FAR is the most important evaluation for all generative outcomes, but base on a generative result, it is easy for computer to calculate. FAR can be determined completely by the follow equations:

$$FAR = \frac{\sum_{i=1}^{n} Math.floor(\frac{buildingHeight_i}{storeyHeight_i}) \times floorArea}{siteArea}$$

Comment: *Math.floor ()* is a method of class of "*Math*", which returns the largest integer less than or equal to the result of "*buildingHeight/ storeyHeight*".

The generative achievement possesses the highest FAR is saved in a computer program structure of arrays.

"*Occlusion Rate*": The lower, the better.

Many other factors, such as orientation of landscape, public space and so on, will influence layout of buildings. In this case, the river to the west of site is such kind of factor needed to be considered. With the control of "O*cclusion Rate*", "HighFAR" can figure out which layout of building will make an optimized generative outcome. Compare with sole goal of FAR, "*Occlusion Rate*" is more complex. For example, for global optimization, the tool must ensure that most of buildings could view the river to the west. Furthermore, it must be a scientific calculation. As a method, the tool set rules as follows:

(i) Each agent calculates the number of buildings which block their views to the river within 30 degrees sight scopes, and obtain the integer of "*blockNumber_i*" for each building.

(ii) Accumulating total number of all buildings which block their views to the river within 30 degrees sight scopes, and obtain the integer "*totalBlockNumber*".

(iii) After the above two steps, we can get a summation value of "*blockNumber / totalBlockNumber*" of all buildings, which means the block rate of current layout. Save the highest value in array as the "*Occlusion Rate*", but for the evolution of the tools, a layout with lowest "*Occlusion Rat*e" will be the best solution.

The ideal layout with low "*Occlusion Rate*" is also saved in a computer program structure of arrays.

Synthetic Evaluation: Synthesis of FAR and "*Occlusion Rate*"

Synthetic evaluation factor contains both FAR and "Occlusion Rate" generated by genetic algorithm. With the help of "*weight*", different "*weight*" for FAR and "*Occlusion Rate*" are

set for getting a synthetic outcome. The weights are determined by debugging program for many times. Furthermore, we need to add many other rules for the final ideal generative achievements. Sometimes, the tool can optimize a result with 3.5 or even more FAR, which is a good outcome, but on the other hand, the "*Occlusion Rate*" is too high for many buildings to get ideal view. That's why the synthetic evaluation is needed.

Base on the generative results as a sketch and any other application of software, such as AutoCAD, Photoshop and etc., it is a simple work for architects to implement the remained works of architectural drawing. They are presented in figure 8.

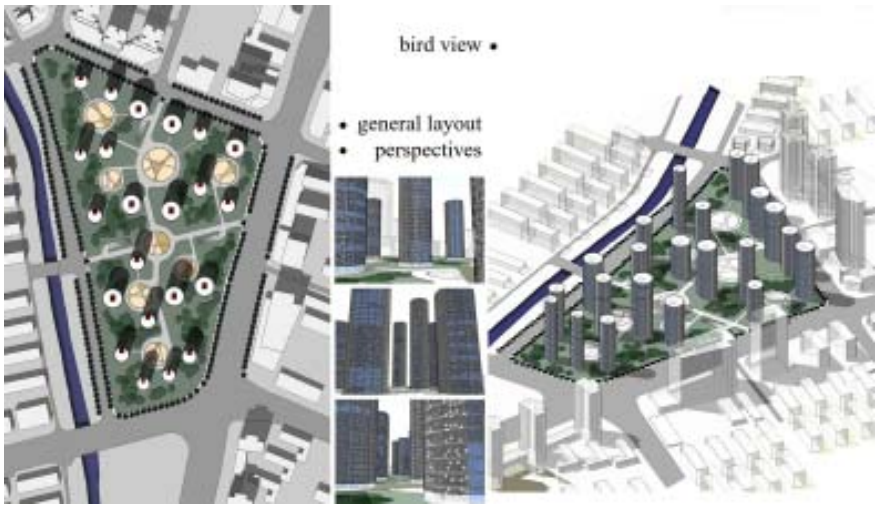

Figure 8: Final achievement

5. Further Work of the Tool

The development of the generative tool employed architectural rules, mathematics, physics, and even knowledge of astronomy, which reflects generative design is an interdisciplinary integration and innovation. From view of computer program algorithm, multi-agent system and simple genetic algorithm are key contents for developing of "HighFAR".

However, there are any development spaces when it is applied to practical application. Firstly, normally, the layout of building site usually involving kinds of architectural plan layout demanded rather than only high-rise buildings, which from the angel of architect may the same question, but from that of computer science they are completely different problems and compute algorithms. It needs new methods for agents encoding. Secondly, by computer programming and relative algorithm, "HighFAR" shows a new method for architectural design and building layout in China, but for the real estate projects, the plan of building normally can't be strictly with only circular layouts, which make "HighFAR" requires more demand-oriented quality evaluation. Another important step will be search ways for real genetic algorithm rather than simple genetic algorithm, and make the tool possess more efficient evolution.

A powerful generative application software focused on high FAR residential design and avoid the defects presented above is being developed recently in our group, and the powerful generative tool will apply to real estated practice in China.

References

F. Scheurer: 2005, *A Simulation Toolbox For Self-Organisation In Architectural Design*. In: S. Sariyildiz, B. Tuncer (Eds.), Innovation in Architecture, Engineering and Computing (AEC), Vol. 2, Delft University of Technology, Faculty of Architecture, Rotterdam, NL, , pp. 533-543.

F. Scheurer: 2005, *Turning the Design Process Downside-Up - Self-organization in Real-world Architecture*. In: B. Martens, A. Brown (Eds.), Computer Aided Architectural Design Futures, Springer, Dordrecht, Vienna, 2005, pp. 269-278.

Jeffrey Krause, BArch USC: 2003, *The Creative Process of Generative Design in Architecture*. GA2003 - 6th Generative Art Conference / Exhibition / Performances. Milano, Italy. http://www.generativeart.com/.

LiBiao: 2007, *A Generic House Design System: Expertise of Architectural Plan Generating[C]*. In: YuGang, ZhouQi, DongWei eds, Proceedings of the 12th International Conference on Computer-Aided Architectural Design Research in Asia (CAADRIA). Nanjing, China.

N. Biloria: 2007, *Adaptive Corporate Environments[C]*. In: YuGang, ZhouQi, DongWei eds, Proceedings of the 12th International Conference on Computer-Aided Architectural Design Research in Asia (CAADRIA). Nanjing, China.

Paul S. Coates AA Dipl: 2004, *some experiments using agent modelling at CECA [EB/OL]*. GA2004 - 7th Generative Art Conference / Exhibition / Performances, Milan Italy. http://www.generativeart.com/papersGA2004/22.htm.

F. Scheurer: 2003, *The Groningen Twister An experiment in applied generative design [EB/OL]*. Switzerland: Proceedings of Generative Art 2003 conference, www.generativeart.com.