INTELLIGENT MULTI-OBJECTIVE OPTIMIZATION METHOD FOR COMPLEX BUILDING LAYOUT BASED ON PEDESTRIAN FLOW ORGANIZATION

A case study of People’s Court building in Anhui, China

YUNQIN LI¹, JIAXIN ZHANG² and CHUANFEI YU³
¹,²,³School of Architecture, Southeast University
¹seraphliyq@gmail.com ²,³{220160132|yuchuanfei}@seu.edu.cn

Abstract. The pedestrian flow of the building influences and determines the layout of the building’s plan. For buildings with complex flow such as courts, airports, and stations, mixed flow line and low traffic efficiency are prone to be problems. However, the optimization of the layout of complex flow buildings usually relies on the architect’s experience to judge and trials to improve. To overcome these problems, we attempt to establish a parametric model of buildings’ plan (taking a typical court building as an example) with information about the different pedestrian flow and functional groups. Based on the Rhino and Grasshopper platform, we take the minimum of different pedestrian flow path length and the maximum of total spatial integration value and the minimum of total spatial entropy value as the starting point, combines pathfinding algorithm, Space Syntax and multi-objective genetic algorithm to optimize space allocation. The result shows that, compared with the original scheme, the intelligent optimised scheme can reduce the spatial waste caused by improper flow organisation, effectively improve space transportation capacity and spatial organization efficiency.

Keywords. Intelligent optimisation; space allocation; multi-objective optimization algorithm; Space Syntax; pathfinding algorithm.

1. Introduction

Layout planning is an important sub-category of computational design which is an important issue and a challenging task in building intelligent optimisation design in this digital age. One of the important parts of the layout planning is pedestrian flow organisation design which is also the essential core issue of architectural tasks. The design of the majority of current building layout is led by aesthetics and prescriptive regulations, while the prescriptive regulations have derived from building evacuation strategies. A less than perfect situation exists at present as free, uncongested flow within built spaces is not only an issue of safety and security.
but also of comfort and efficiency. Quantitative analysis of pedestrian flow by establishing a parametric building model can help architects optimise space layout, design and management. This can result in cost savings in terms of optimised design, and solutions that improve user comfort, safety as well as efficiency.

In the current state of art, the majority of existing plan layout models are driven by analysis rather than computation, especially for the optimisation under pedestrian flow organisation. There are two main reasons for this. One, most of the existing layout optimisation problems are mostly in the qualitative research stage, lacking the use of systematic analysis methods for quantitative layout. The majority of existing crowd flow simulation models which can help quantitative analysis of layout planning have evolved from building evacuation needs. Such scenarios require only an assessment of the time the occupants take to exit the buildings, usually using the nearest exits. These tools can be used to analyse a given design within a given scenario, but it is not possible to further simulate and optimise other conditions beyond the evacuation specification. The second reason is the complexity of the models. People movement models tend to be inherently complex, as they need to account for a combination of factors such as spatial configuration, population behaviour, and agent interaction. When the architects are in the spatial layout, it is difficult to grasp the movement behaviour of a large number of people, and architects can only adopt the top-down approach, relying on intuition and experience to judge the spatial form, and rarely use the accurate, objective and rational way in building layout. The optimisation process is time-consuming, and it is difficult to evaluate and compare the optimization results accurately.

Until now, some scholars have made some useful explorations on the issue of architectural design optimisation. Reported attempts to optimise the process of layout design started over 50 years. Researchers have used several problem representations and solution search techniques to describe and solve the problem. In the 1960s, American engineers Souder and Clark inhabited intelligent algorithms in “Computer Technology: A New Tool for Planning” to optimise the hospital building space layout according to the patient’s shortest walking route. Based on the Grasshopper platform, Lu Shuai combined spatial syntax theory and the Galapagos algorithm (multi-island genetic algorithm) to optimise the total connection value and the depth value of the functional group. By optimising the optimal position of the functional group, a layout prototype of a complex pedestrian flow buildings such as the airport was generated. Wang Wei used the metball algorithm, Voronoi algorithm and four eucalyptus algorithm in the Grasshopper platform, to analyse and optimise traffic space accessibility, shop uniformity and commercial building experience for the layout optimisation problem of commercial buildings which is of large traffic volume. And the optimisation results were graphically visualised.

It can be seen from the above research that the starting point and optimisation method of each research are different. At present, the optimisation of the building layout under the guidance of complex pedestrian flow organisation such as court buildings remains to be studied. This paper is aimed at putting forward an intelligent method based on a parametric model, pathfinding algorithm, Space
Syntax and multi-objective optimisation algorithm to optimise spatial allocation for buildings of complex pedestrian flow. This method can simulate different pedestrian flow, quantitatively calculate and evaluate spatial layout with visually optimised output results which are more accurate and intuitive.

2. Related Work and Background Concepts

Building layouts are mostly created as a result of an iterative trial-and-error process that needs substantial expertise and considerable amount of time. For buildings with the complex pedestrian flow, spatial planning becomes a cumbersome task due to the increase in the number of design factors, which makes it more reasonable and practical to cite relevant quantifiable analytical methods and use specific software. In the current court model, the strict rules order makes the court space closed. If the parties want to participate in the trial, they must follow the judicial procedures to complete the activity procedures in the court square, the filing hall, the party reception room, other functional rooms, the etiquette hall, the waiting hall and the court, one by one. The court flow design also strictly reflects the sequence of this activity procedures (See Figure 1). Although this strict flow design can ensure that different groups of people could complete the “right” things, different people are unable to move freely as they wish, which makes the flow of court space prone to be monotonous and inefficient. It is also the reason why we cannot utilise the time we queued in the petition lobby, and it means that the court building space is not open enough for people and not popular enough for humanisation. Therefore, with the aim of the court’s layout optimisation which is to improve the traffic flow efficiency, the space should have the closest connection and the most common reachability. The path length of different people should be reduced to ensure that different pedestrian flow between spaces can be more efficient which can improve the efficiency of the trial work. For the above optimisation objectives, we have adopted three methods, space syntax, pathfinding algorithm and multi-objective optimisation, which will be discussed in the following subsections.

Figure 1. Internal traffic flow organisation of different crowds in court buildings.
2.1. SPACE SYNTAX

In order to ensure that the space has the closest connection and the most common reachability, it is necessary to quantify “connectivity” and “reachability.” In this design, we chose the concept of “integration value” and “Entropy value” in space syntax to quantitatively describe “connectivity” and “reachability.”

The theory of Space Syntax was initiated as a theory of architecture, seeking to explain the meaning of spatial configurations as to their social functions. Although it has been mostly used in urban analysis, it is still an architectural theory, and its basic examples are architectural. In simple terms, the theory of space syntax is focused on how spatial units relate to one another in buildings and built environments.

Integration is a measure of centrality that indicates how likely it is for a space to be private or communal. The more integrated a space, the shallower it is to all other nodes in a configuration, the more for a space to be communal. Entropy in Space syntax is a measure of the distribution of locations of spaces in terms of their depth from a space rather than the depth itself. If many locations are close to a space, the depth from that space is asymmetric, and the entropy is low. Intuitively, the higher the entropy value, the more difficult it is to reach other spaces from that space and vice-versa. Therefore, to have the closest “contact” of the space and increase the space accessibility, we can seek the maximum value of the integration value and the minimum value of the entropy value. Today, tools of space syntax are relatively mature, such as Depthmap, Confeego and other tools have been widely used. In this paper, we use the Space syntax toolkit which is based on the Grasshopper parameterisation platform to analyse the Integration value and the Entropy value.

2.2. PATHFINDING ALGORITHM

In order to improve traffic efficiency, this paper attempts to optimise the length and mode of traffic path of each group, which involves the problem of the pathfinding algorithm and shortest path. The shortest path search is a classic algorithmic problem that seeks to find the shortest path between two nodes in a graph consisting of nodes and edges. As a representative algorithm of the pathfinding algorithm, A* algorithm can solve the path-finding problem in architectural layout and provide a reference for plan layout optimisation. Basically A* algorithm works based on grids called “Nodes”, every time the algorithm search for adjacent nodes and evaluate the nodes according to a key function \( F(n) = G(n) + H(n) \), where \( G(n) \) is the movement cost to move from the starting node to given node \( n \), and \( H(n) \) is the estimated movement cost to move from node \( n \) to the final destination, which is called heuristic function. The node moves from a start point to an adjacent least \( F \) value node until reaching a destination; then a path is generated.

Based on the A*algorithm and some sample codes, we use AstarPathFinder, a grasshopper component, which just needs the start point, the destination point and obstacles to find path autonomously. Inside this component, simply speaking, the space including two points and obstacles are subdivided into voxels, as “nodes” for A*, and voxels which collide with obstacles are defined non-walkable. Therefore, we can regard the function rooms as obstacles, set the starting point and the end point according to the functional sequence of different pedestrian group, and then
run the program to simulate the motion path of each type of pedestrian flow and record the result.

2.3. MULTI-OBJECT OPTIMISATION

In general, the optimisation problem refers to obtaining the optimal solution of the objective function by a certain optimisation algorithm. The single-objective optimisation problem means that there is only one objective function for optimisation, and when there are two or more optimisation objective functions, it becomes a multi-objective optimisation problem. In this paper, we need to optimise the integration value and entropy value of the plane and the path length of each pedestrian flow. Therefore, the multi-objective optimisation method is needed.

Octopus is a multi-objective optimisation plugin based on Grasshopper platform, which combines Pareto optimal principles with evolutionary algorithms. It provides a wealth of customizable optimisation options, providing rich customisation and parameter options for multi-objective optimisation problems as well as a rich set of algorithm parameters (including the SPEA-2 and HypE algorithm). It also offers an intuitive and effective visual feedback interface, rich interactive means, and convenient data storage transmission. Compared with Galapagos (Single-objective optimisation component in the Grasshopper platform), the algorithm of the Octopus is more scientific and more interactive. What is the most important is that it provides a quick multi-objective optimisation service for ordinary designers, which brings more possibilities for the layout optimisation problem.

Figure 2. A schematic diagram of intelligent multi-objective optimisation design.
3. Optimisation workflow

After the analysis and rewriting described above, we have obtained the optimisation rules for the architectural space that can solve the layout optimisation problem of court buildings which are examples of complex pedestrian flow buildings. Combined with the specific design requirements of one of the People’s Court building in Anhui, China, the following workflow corresponding to the optimisation rules are used to complete the layout optimisation. Figure 2 illustrates an overview of the optimisation workflow, which consists of four main steps.

3.1. STEP 1 PREPARING THE INPUT

Designers start with extracting the gravity centre of the functional group from the graphics that can be used for spatial syntax analysis in the original plan, and lists the corresponding functional group area lists and their spatial label (name) list. According to the requirements of the design principle of the court and the topological relationship between functional groups, the designer draws a line between the centre of gravity of the functional group that needs to be directly connected (the circle representing the functional space), that is, the plan topological relationship diagram is obtained. An example of an input plan can be seen in Figure 3, while Table 1 shows an example of input parameters.

With the definition of integration value and entropy value, we have quantified the spatial “contact degree” and “accessibility”. In order to make the space have the closest and the most accessible “contact”, we use the Octopus tool in the Grasshopper platform to optimise the integration value and the entropy value. Taking the optimization of the first plan as an example, according to the functional analysis and design task requirements, there are four functional groups in the airport design, including petition, archives, office and others. By extracting the position coordinates and adjacency relationships of the functional groups in the original design drawings, we can obtain the relevant integration values and entropy values in conjunction with the Spacesyntax tool (see Figure 4). Furthermore, we can write a program in the Grasshopper platform to calculate the sum of the distances between the groups with the integration value as the weight, that is, the total connection value of the space. Similarly, the entropy value of the space can also be calculated in the Grasshopper platform.

![Figure 3. Input shapes of the first experimental optimisation (First Floor Plan).](image-url)
3.2. STEP 2 OPTIMISING THE INTEGRATION VALUE AND THE ENTROPY VALUE OF FUNCTIONAL GROUPS BASED ON SPACE SYNTAX THEORY

Next, we take the total integration value and the total entropy value as the fitness values, and take the maximum value of the total integration value and the minimum value of the total entropy value as the target value. In this way, the multi-objective optimisation is performed by the Octopus tool method (In Octopus, the red arrow points to the argument and the green arrow points to the target value). After 95 generations of calculations, the test reached a steady state. Extracting the spatially disintegrated groups of the 3rd, 11th, 35th, and 95th generations with typical changes, it can be observed that the individuals of each generation gradually converge from the disordered discrete state to a Pareto frontier boundary. We can choose the appropriate solution from the elite solution to carry out the following optimization. At this time, the function group position we chose ensures that the space has the largest “total integration value” and the “smallest total entropy value”, which means that the space has the closest “contact” and the most common “accessibility” (see Figure 5). Subsequently, the optimised functional group position is projected into the plan by the Flow AlongSurface operation in Rhino, and the optimised centre of gravity position of the functional group can be selected.

Table 1. Names and areas of functional groups.

<table>
<thead>
<tr>
<th>Names of functional sub-areas</th>
<th>Archive</th>
<th>Archive</th>
<th>Device</th>
<th>Mental</th>
<th>Meeting</th>
<th>Device</th>
<th>Filing</th>
<th>Archive</th>
<th>Office</th>
<th>Media-</th>
<th>Exhibition</th>
</tr>
</thead>
<tbody>
<tr>
<td>Number of functional sub-areas</td>
<td>1</td>
<td>2</td>
<td>3</td>
<td>4</td>
<td>5</td>
<td>6</td>
<td>7</td>
<td>8</td>
<td>9</td>
<td>10</td>
<td>11</td>
</tr>
<tr>
<td>Area (m²)</td>
<td>81.6</td>
<td>103.4</td>
<td>103.4</td>
<td>39.8</td>
<td>81.6</td>
<td>97.4</td>
<td>41.6</td>
<td>41.6</td>
<td>84</td>
<td>112</td>
<td>112</td>
</tr>
</tbody>
</table>

Figure 4. Calculate the integration value and the entropy value with the Spacesyntax tool.
3.3. STEP 3 OPTIMISING THE SHORTEST PATH OF DIFFERENT FLOW ACCORDING TO A* ALGORITHM

After obtaining the optimised specific position of gravity centre of the functional group’s, activities and path lengths of different types of crowds can be simulated. In this paper, the AstarPathFinder tool based on A* algorithm is used to send different pedestrian flow represented by unit individuals from different entry positions into different spaces. Each individual generated according to probability distribution and contains the functional sequence information (that is, activities that individuals need to complete in order after entering the space, such as “petition → archives → office” is a possible sequence of functions). After entering the space, the individual selects one of the functional groups in turn according to the requirements of the functional sequence (see Figure 6). According to the area index requirements of the task book and the original design drawings, if the functional group area and the centre of gravity are constant, the variable functional group can be used as an obstacle by changing the length and width of the functional group, and thereby different paths can be simulated. At the same time, according to the traffic volume of different groups of people, the program can calculate the weighted total pedestrian path length and optimise its minimum value by Octopus plugin. After multiple iterations, different optimisation results can be obtained, and We select the appropriate solution that meets the design requirements for the next step optimisation. Therefore, the basic situation of crowd activities and functional groups in space can be mastered as the basis for layout optimisation.
3.4. STEP 4 COMPARISON AND EVALUATION OF THE OPTIMISED LAYOUT RESULTS

Although the layout optimisation prototype obtained in the previous step is generated based on pedestrian flow organisation optimisation, the spatial allocation remains on the stage of functional groups. It means that we should deepen the prototype results, for example, dividing the room, planning the traffic space such as the corridor and adjusting the location of different entrances for different pedestrian flow. Therefore, we select one of the building layout optimisation results we obtained in the previous step for deepening research according to the original design requirements (see Figure 7). It can be seen that the total weighted path length of the different pedestrian flow after optimisation is 264 meters, while the figure before the optimisation is 339 meters. The total Integration value of the space is relatively increased while the total Entropy value is relatively reduced. In this way, it can be proved numerically that the traffic efficiency of the space is improved, the spatial accessibility is enhanced, and the space waste is reduced due to improper organization of the different pedestrian flow.

4. Conclusion

By the combination of Space Syntax, pathfinding algorithm, multi-objective optimisation algorithm and parameterised platform, we build an intelligent optimisation platform for the multi-objective layout of complex pedestrian flow buildings based on Grasshopper visual programming. Based on the principle of “simulation-calculation-optimisation-evaluation”, the building traffic efficiency and plan accessibility are improved. The whole intelligent optimisation process of visualisation and controllability makes it easier for designers to find the optimal layout plan through human-computer collaborative decision-making, which can provide reasonable, scientific and decisive support for layout optimisation of complex flow line building.

However, there are still some problems with this optimisation method. First, due to the limitations of theories of space syntax, it is difficult to accurately analyse and optimise the layout of the three-dimensional spatial building, especially for those with relatively complex vertical space. Also, the simulation of different pedestrian flow paths only consider the objective level of path length calculation and ignore the individual’s behaviour patterns and their psychological feelings. Last, there are fewer effective solutions in the current optimisation process which can be used in the next step to deepen the design, and the designer still needs to
adjust the following design details manually. These are the core issues that need to be addressed in subsequent research.

![Figure 7. Layout optimisation result.](image)

**References**


