

Inventing Circulation Patterns using Available Metaheuristic Solvers

From ambiguous fitness functions to procedural patterns

*Richard Schaffranek
Vienna University of Technology, Austria
<http://www.gbl.tuwien.ac.at/Archiv/digital.html>
richard.schaffranek@gmail.com*

Abstract. *The way people navigate through spaces has been studied for quite a while. Different models have been described and validated through empirical studies. This paper explores the use of such models in an ‘inverse design’ (Faucher and Nivet, 2000) process, applying available metaheuristic-solvers. More detailed, it showcases the description of an ambiguous fitness function as base to invent a new circulation pattern. As such technics are time consuming compared to the use of readily available patterns, an attempt is made to analyse and understand the invented circulation patterns, in order to come up with a procedural algorithm that would generate circulation pattern with the same characteristics.*

Keywords. *Procedural; circulation; inverse; metaheuristic; prototypical.*

INTRODUCTION

The way people navigate through a city, a building, a spatial arrangement is determined by the arrangement itself. Different methods have been made available, and validated through empirical studies, in order to understand and simulate different behaviours of people (e.g. way finding strategies (Hochmair and Frank, 2000), configurational relationships between spaces (Turner, 2007)). As exciting these methods are, very little new has been invented with their help. In design processes they sometimes get utilized in order to understand the impact of different proposals, but “designers often refer to regulations and guidelines such as urban codes or pattern books While such an approach is useful as it ensures a certain standard in the planning of environments, it is relatively inflexible in its

ability to respond to changing context” (Schneider and Koenig, 2012). In order to overcome this, Schneider and Koenig demonstrate an approach that was described by Faucher and Niver (2000) as “inverse design.” (“Inverse design” can be referred to as design by intents, used as constraints to a form and not as form by itself. Schneider & Koenig are focusing on visuospatial properties, using isovists, to demonstrate the possibilities of an “inverse design” approach in their paper.)

Such an approach has a lot of potential, but if the objectives are clearly defined (e.g. generate a circulation, where the distances between all pairs of points is a minimum), the outcome is rather predictable. Further it is time-consuming, compared to the use of patterns. This paper describes a way, how to

use an ambiguous fitness function with an available metaheuristic-solver (multiple objectives, simplifications and combinations of the different methods mentioned above) to invent a new procedurally generated circulation pattern. In order to showcase the possibilities the task to generate a labyrinth, the most generic form of circulation, which allows for getting lost, was chosen. (Metaheuristic-solvers can be used to solve optimisation problems without knowing specifics about the problem. Therefore they can be applied to a variety of problems. Depending on the type e.g. a genetic solver, different strategies e.g. evolutionary principles are applied to search through possible solutions, trying to find the best solution. The quality of a solution is defined by a single value, its fitness. Finding the best solution is not guaranteed. Through plugins to existing CAD programs e.g. Grasshopper3d, they are made available to a broad audience.)

BEHAVIOURAL MODELS

Procedural Generated Labyrinths

Different algorithms, like cellular automata's or adaptations of breadth first search can be used to procedurally generate labyrinths. At first these generated circulation patterns look very complex. When looking at the underlying structures, they are not complex, but only complicated. This can be understood when looking at the traversal tree (including the back edges, cross edges and forward edges), referred to as justified graph (Hillier and Hanson, 1988), of such a labyrinth. These methods only generate tree like structures, which, even when only experienced from a first person perspective, can be solved with ease. A simple rule "keep the left hand on the wall and start walking" will guide you through safely. The first objective, as a result of this observation, is that a labyrinth allowing for getting lost has to be constructed as a network. This might seem like a contradiction at first (Equation 3).

Way finding in Unknown Networks

A way to describe how people navigate through un-

known networks on a local scale is the least-angle heuristic (Hochmair and Frank, 2000). If the direction to the destination is known, in the choice between different directions, the direction that deviates the least is the most obvious choice. Or: Walk in the direction where the Euclidian distance is shortened. The negation of such a simple way finding heuristic is the second objective in the definition of a fitness function (Equation 2).

Simplest Path

When looking at the choice of path between A and B in a network, it can be described by the angular segment analysis (Turner, 2007). In this case the shortest path is not described by the distance but by the sum of changes in direction (angle) between A and B. This concept will be used in order to measure the complexity of the labyrinth on a global scale (Equation 1).

SIMPLIFIED MODEL

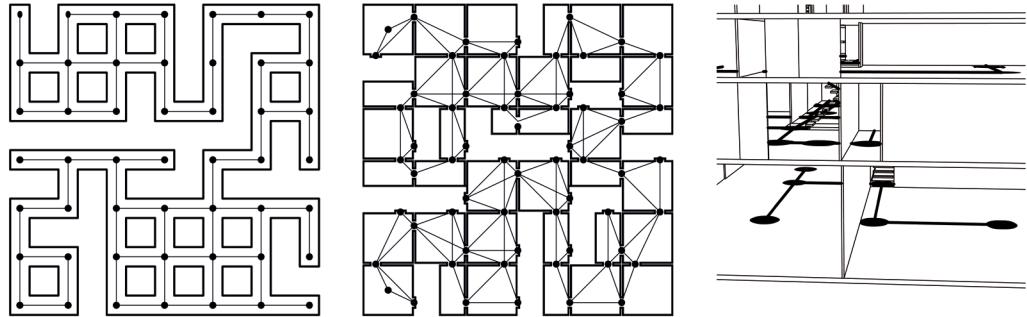
The described models operate on very specific forms of representation of a given environment. For the calculation of the simplest path Turner (2007) proposes road-centre lines for an urban environment, while the model of Hochmair and Frank (2000) is described by vectors operating within a continuous representation of the surrounding. Further to use such models with a metaheuristic-solver the representation has to be generated automatically, from a given spatial configuration / the spatial configuration has to generate automatically from a given representation, to search through different solutions. This duality is referred to as prototypical model in this paper.

As can be seen in Figure 1 (right), a simple grid-based graph, only allowing for perpendicular movement, represents a three-dimensional layout. Such a simplification cannot be used for detailed analysis but seems admissible for the use in combination with a metaheuristic-solver. Speed is of essence.

AMBIGUOUS FITNESS FUNCTIONS

To use the stated objectives as a fitness function a mathematical description is needed. Let $G(V, E \in G)$ be the Graph representing the Labyrinth and $p = (P_0$

Figure 1
Different prototypical models:
graph representation and
different spatial layouts.



$= A, P_1, P_2, \dots, P_n = B; P_{0,n} \in V(G)$ the shortest path from A to B.

The costs $c(p)$ (sum of changes in direction) for a trip, along the shortest path, from A to B must be as high as possible. The more the shortest path p converges to a space-filling curve the higher the costs $c(p)$ (Figure 2).

$$c(p) = \sum_{i=1}^{|p|-1} \text{angleBetween}\left(\overrightarrow{p_{i-1}p_i}, \overrightarrow{p_i p_{i+1}}\right) \quad (1)$$

To negate the way finding heuristic described by Hochmair and Frank (2000), the average of Euclidian distances $d(p)$ from all nodes part of the shortest path p to A and B must be as high as possible (Figure 3).

$$d(p) = \frac{\sum_{i=0}^{|p|} \text{dist}(p_i, B) + \sum_{i=0}^{|p|} \text{dist}(p_i, A)}{|p|} \quad (2)$$

In order to create a network-like structure, the more edges not part of the shortest path from A to B, the better (Figure 4).

$$n(p) = |E(G)| (E \setminus V_1 \notin p \vee E \setminus V_2 \notin p) \quad (3)$$

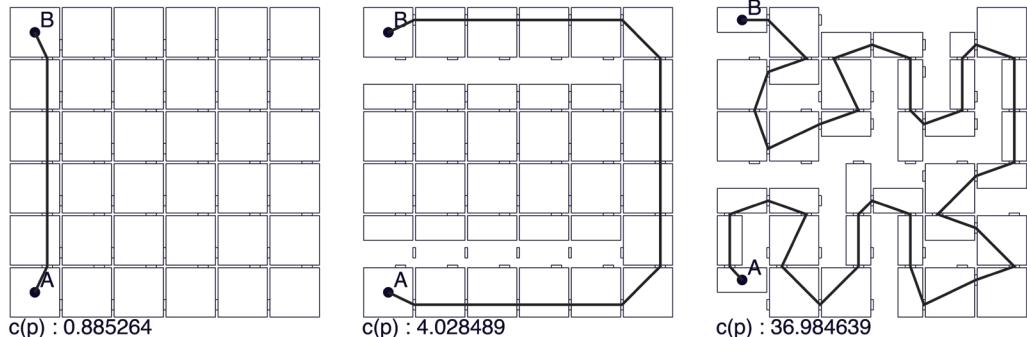
As each objective is clearly defined, the results for each objective alone are predictable. Only when combined the result gets ambiguous since the different objectives contradict each other. As stated before if maximizing the sum of changes in direction $c(p)$, the path will result in a space-filling curve. Contrary the amount of edges not part of the shortest path $n(p)$, would get minimized. To combine the different values to a single fitness f , the product of the three values is calculated.

$$f = c(p) * d(p) * n(p) \quad (4)$$

For all solutions without a path between A and B, f will result in 0, making them indistinguishable, even so only a little change to one of these solutions might result in the best solution found so far.

To overcome this problem a discontinuous fitness function is proposed. This enables the algorithm to search for different properties in its run. As long as there is no walkable path between A and B,

Figure 2
left: The path p is close to a
straight line, the resulting
costs $c(p)$ are a small;
middle: The path p has a C-shape,
costs $c(p)$ increase;
right: The more the path converges to a
space-filling curve, the higher
the costs $c(p)$ get.



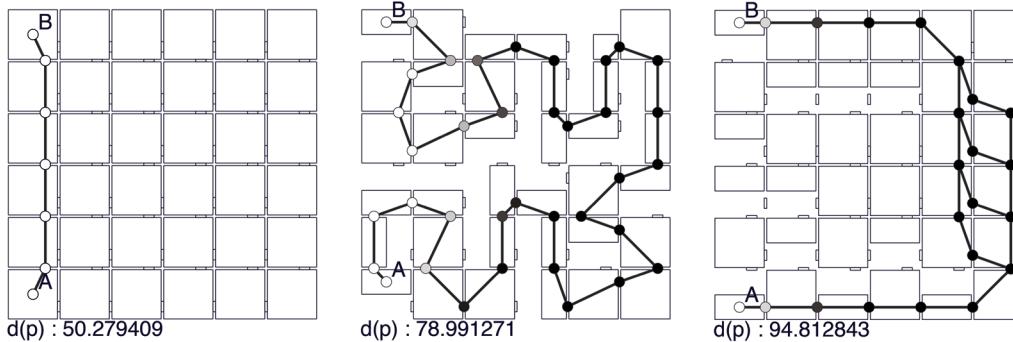


Figure 3
left: The path p is close to a straight line, the resulting costs $d(p)$ are small; middle: The path with the highest $c(p)$ from Figure 2 only performs average when it comes to $d(p)$; right: The more parts of the path are far away from A/B the higher $d(p)$.

the fitness is calculated by f_1 , ensuring that solutions containing more edges are rated better, since they are more likeable to result in a viable solution. As soon as a path between A and B is found, the fitness is calculated according to f_2 , searching for the properties described above.

$$f_1 = |E(G)| \quad (5)$$

$$f_2 = c(p) * d(p) * n(p) \quad (6)$$

The labyrinths found by this ambiguous fitness function strongly differ to other labyrinths. Movement is not restricted by dead ends. When looked at it, from bird's eye view, the solution is found quickly, but seen from first person perspective it is highly disorientating.

PROVING ASSUMPTIONS

In order to understand the form characteristics of the generated labyrinths and to prove the made as-

sumptions, some generated labyrinths were used as levels for a casual game. (Computer games have been successfully used in other fields. The computer game 'Fold It' utilizes the intuitive human mind for 'solving' three-dimensional puzzles (proteins) for biochemists. When solving the puzzle, the player actually designs a three-dimensional geometry fulfilling given constraints. Parallels to architecture are accidental [1].)

The player finds himself at the starting point A of the labyrinth (first person perspective) and has to find the exit B. The only orientation help for the player is an arrow pointing straight towards the exit and the sky, which is shaded from deep blue to light grey (Figure 5). The shading of the sky doesn't depend on the position of the player while the red arrow changes depending on the position. While exploring the labyrinth the player drops breadcrumbs (sketched

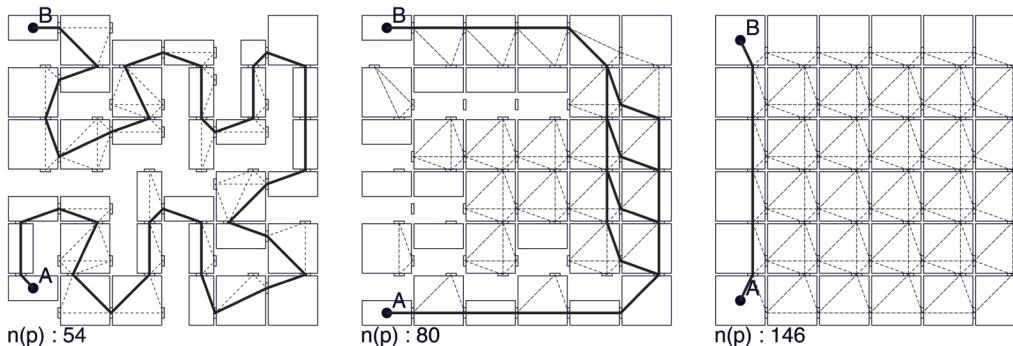
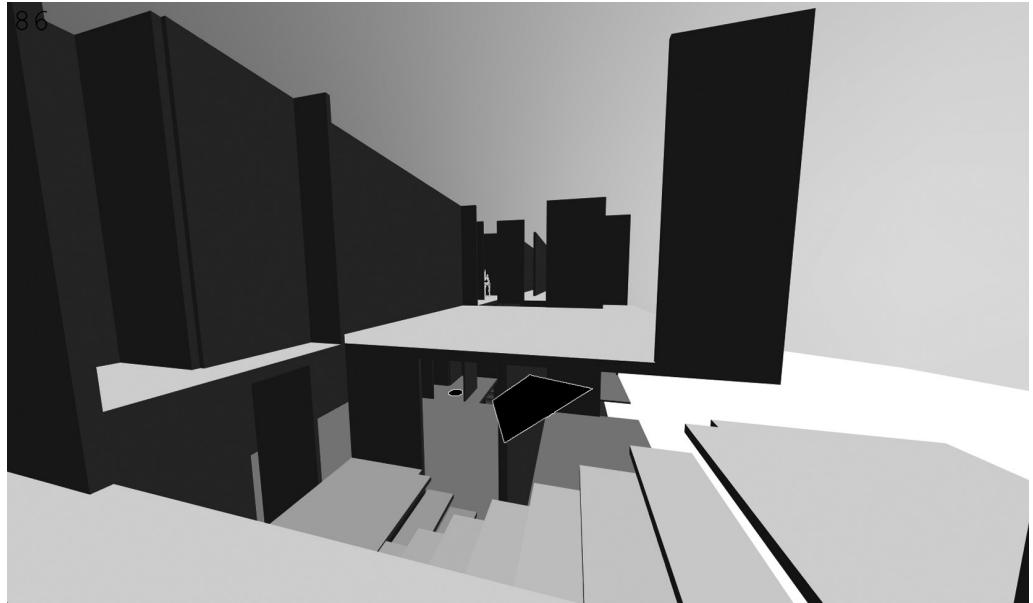


Figure 4
left: high $c(p)$ value; middle: high $d(p)$ value; right: The path p is close to a straight line, a lot of lines are not part of the shortest path from A to B, the resulting costs $n(p)$ are high.

Figure 5
First person perspective of the level 'Maze3d'; the red arrow in the foreground is pointing towards the red dot marking the exit; The game Labyrinth Runner can be played under [2].



people). A player has enough breadcrumbs to walk 3.33 times the length of the shortest path. The game ends, if the exit is found or the player runs out of breadcrumbs, in this case the player is considered lost within the labyrinth. The positions of the breadcrumbs are recorded and stored in a database to trace the players.

As the simplest notion, the dataset collected through the game can prove the assumptions described by the ambiguous fitness function. For a simple level e.g. 'RoomToRoom 6x6' (two-dimensional, small number of different rooms) from 341 finished plays between the 14.01.2012 to 28.05.2013, 35 players did not find the exit. The average score was 542 of 1247 possible, the median score was 540. If including also the aborted games, with at least a third of the breadcrumbs used, an additional of 125 players got lost (34%). From 8 plays, of the three-dimensional level 'Maze3d' that were finished, only 2 players found the exit.

Beyond validation, it provides data to understand player's behaviour within a spatial layout that

wouldn't be built in reality. When playing the same level more often, their understanding of the spatial configuration got better with every game played, not only finding a path through the labyrinth, but finding a shorter path.

When visualizing the average movement direction of all players as vector field, it can be seen that in some areas the vector field has a clear direction and in other areas, where the labyrinth can be characterized as network (more than two connections per room), the average direction of a player can't be determined. As shown in Figure 6 (left), this is the area where most players tend to get lost. The link that connects such network areas with the rest of the labyrinth leads in the opposite direction of a straight line to B.

Further it can be observed that if there is a continuous line of sight within the network area across multiple rooms the average movement direction is more likely to be distinct.

These findings are in accordance with the proposed fitness function. In certain areas the superior-

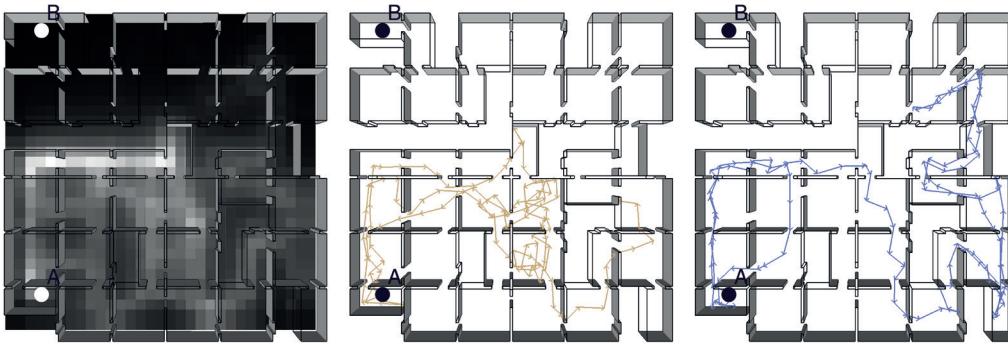


Figure 6
level RoomToRoom 6x6; left: usage by all players that did not find the exit (white: often; black: seldom); middle: first run by Ikagura; right: second run by Ikagura.

dinate circulation converges to a space-filling curve (compare Figure 7 (right) with Figure 2), while the overall C-shape of the circulation can be understood through the assumptions on way finding (Figure 3, Equation 2), the two large clusters fulfil the requirement of a network like structure formulated through equation 3.

PROCEDURAL CIRCULATION PATTERN

As shown so far, an ambiguous fitness functions can be used to invent circulation patterns, but as stated in the beginning such a process is time-consuming

compared to the use of patterns readily available in pattern books like the “Neufert-Baentwurslehre” or produced through simple algorithms as mentioned in the beginning. The question remaining is: Can the outcome of an “inverse design”-approach be described by a (procedural) pattern not only for a specific situation but also in general?

As argued by Schneider and Koenig (2012), the advantage of an “inverse design”-approach over the use of patterns lies within the adaptability to context, but as demonstrated by Wolff-Plottegg (1996) a design is not a result of the context but a result of

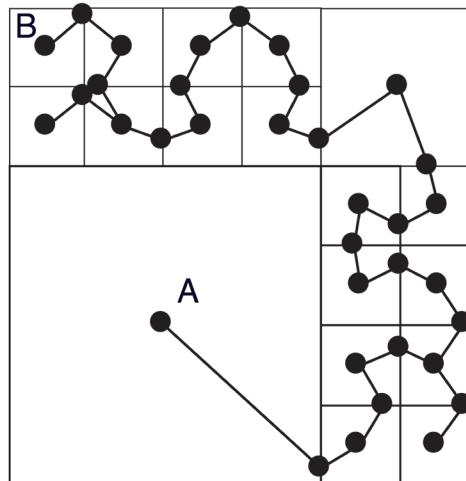
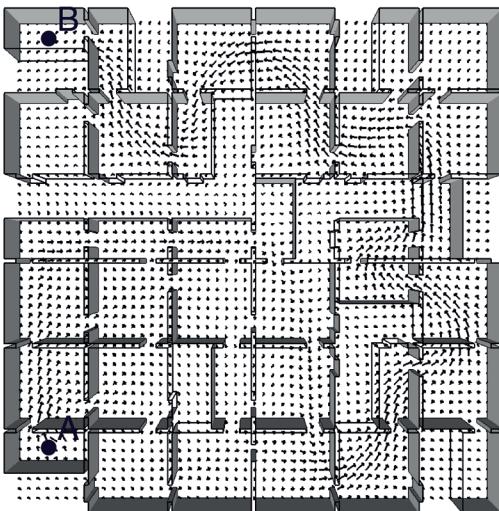
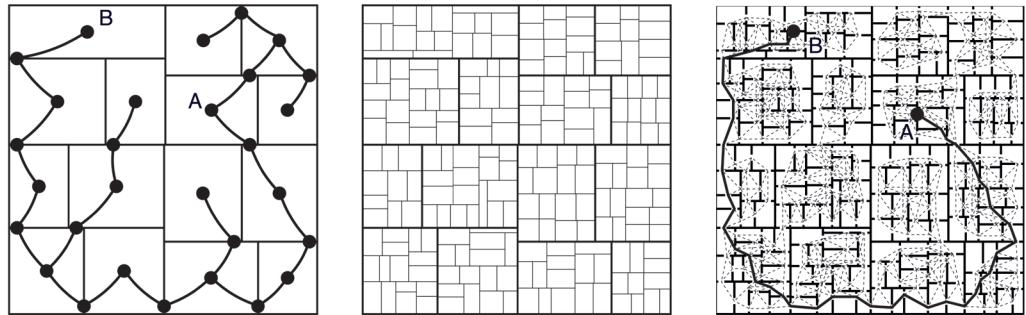


Figure 7
left: The vector field of the average movement direction of ‘RoomToRoom 6x6’; right: super-ordinated circulation pattern, clustering network like areas.

Figure 8
left: primary subdivision with superordinated circulation pattern; middle: secondary subdivision, generating a network within the primary subdivision; right: finished labyrinth layout.



the design algorithm (analogue or digital) itself. So if a procedural algorithm can be found that reproduces the characteristics of the inverse design for a specific case, it should also be able to reproduce the characteristics in general.

The following steps produce a labyrinth with similar characteristics as defined by the ambiguous fitness function:

1. Define an area in which to generate the labyrinth, a starting point *A* and an end point *B*.
2. Create a primary subdivision of the area e.g. kid-tree.
3. Create a graph $G(V, E \in G)$ connecting neighbouring areas. The edge weight of each edge $E = \{e_1, e_2, \dots, e_n\}$ is defined by equation 7.

$$w_i(e_i) = \text{dist}(e_i, v_1, B) + \text{dist}(e_i, v_2, B) \quad (7)$$
4. Calculate the minimal spanning tree of the graph *G*, to get the superordinated circulation pattern.
5. Continue to subdivide the primary subdivision. This can be done iteratively e.g. until a certain minimal room size is reached.
6. To create the secondary circulation pattern, connect all elements of the secondary subdivision within the same primary subdivision to their neighbours.
7. Create final layout.

The resulting layouts are similar to the ones found by the metaheuristic-solver. They can be applied to different context (changing boundary, different starting and end points and even more than two points) without the need to rerun the me-

taheuristic-solver. At the moment this comes with a lot of work analysing, understanding the outcome of the 'inverse design' approach and reformulating it as procedural pattern. Automation would be welcomed.

FURTHER APPLICATIONS

The task of inventing a circulation pattern for a labyrinth was chosen since it is rather clearly defined, yet not as simple as generating a circulation pattern that would minimize the average distance between all areas. This would result in a circulation with an X-shaped staircase (Figure 9).

But through changes to the fitness function other circulation patterns will emerge. This approach e.g. can be used to design a new or alter an existing circulation in such a way, that it would assist users of buildings to meet incidental. Especially in information driven environments this is desirable to facilitate informal communication between users. Further it can help to place attractors for informal communication (such as coffee machines...) at the right places, using a minimal floor area and maximizing effects. As Sailer (2007) suggest, attractors and walking distances have a strong impact on the movement patterns within workplace environments. To invent a new circulation focusing on these characteristics one would need to look for a layout where local centres (minimal distances to all other points in their neighbourhood) are as far away (as equally distributed) from each other as possible (Figure 10).

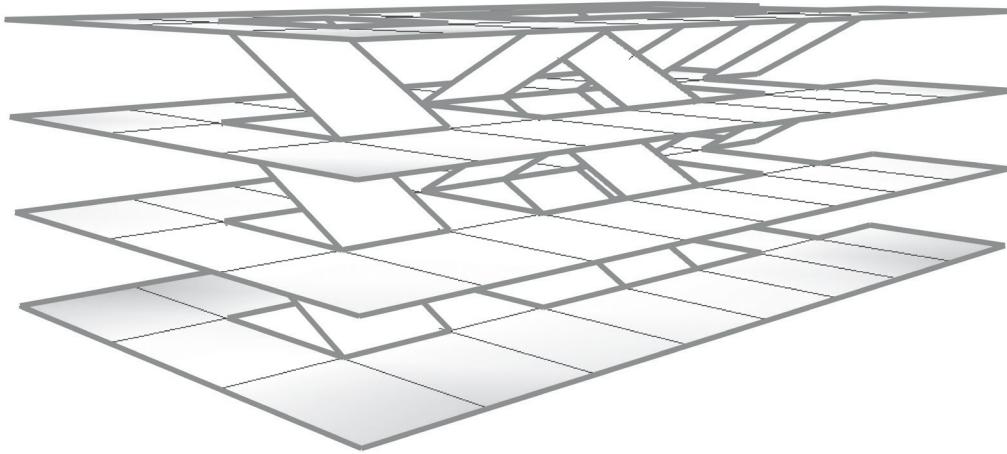


Figure 9
A circulation minimizing the average distance between all areas.

CONCLUSION

The use of computer games to understand the characteristics and validate the success of the generated solution has proven productive in this case. Beyond that other findings could be made through the generated data. Once a player finds a path within an unknown spatial configuration, his understanding of this configuration improves quickly. Further it showed that areas with a network like structure and broken lines of sight are be more disorientating than

dead ends.

Especially through Grasshopper3d, metaheuristic-solvers were made accessible to a broader public of users. As showed, such solvers can be used not only to optimise predictable results (minimize the distance) but also to invent new patterns for architecture. But even though metaheuristic-solvers become readily available, a sensible application of them still needs a lot of in depth knowledge in defining the prototypical model and the fitness func-

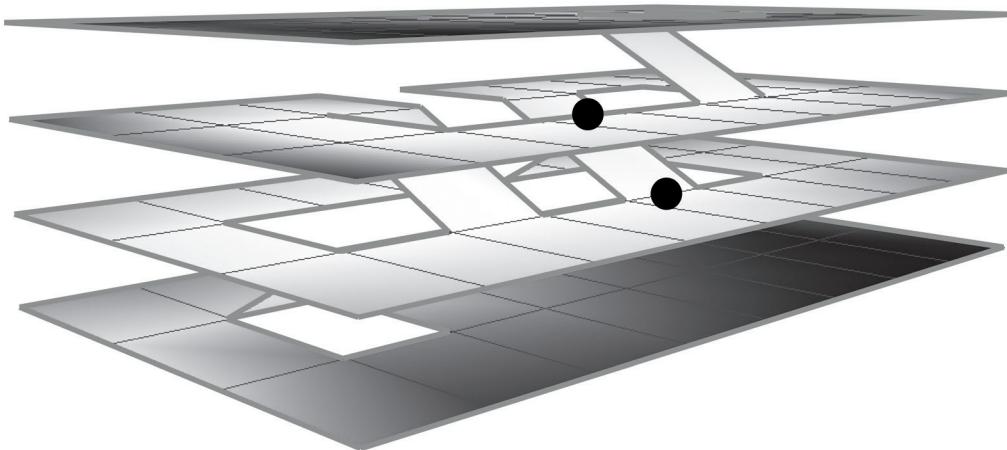


Figure 10
Example with the largest distance between local centres (dots); A double loop emerged as circulation, generating two areas which are closest to all other points than their neighbours; white: smallest avg. distance, black: highest avg. distance to all other points.

tion. Once defined the same fitness function can be applied to different prototypical models and different fitness functions can be applied to the same prototypical model.

As the complete process, of inventing and formulating a procedural pattern, is currently time-consuming, the virtue is, that once defined, the application of it is easy and fast. Further it could be shown that such a pattern can adapt to a changing context.

Even though it was possible, through try and error, to derive a procedural pattern for the stated problem it might not be possible in all cases. A more general automated approach in converting the search result of a metaheuristic-solver into a procedural pattern should be formulated in the future. Such an approach should not start once the result is produced but incorporate data produced during the search process.

REFERENCE

Faucher, D. and Nivet, M. (2000) 'Playing with design intents: integrating physical and urban constraints in CAD', *Automation in Construction*, p. 93–105.

Hillier, B. and Hanson, J. (1988) *The social logic of space*, Cambridge: Cambridge Univ. Press.

Hochmair, H. and Frank, A.U. (2000) 'Influence of estimation errors on wayfinding-decisions in unknown street networks – analyzing the least-angle strategy', *Spatial Cognition and Computation* 2, no. 4, pp. 283-313.

Sailer, K. (2007) 'Movement in workplace environments – configurational or programmed?', 6th International Space Syntax Symposium, Istanbul.

Schneider, S. and Koenig, R. (2012) 'Exploring the Generative Potential of Isovist Fields - The Evolutionary Generation of Urban Layouts based on Isovist Field Properties', 30th International Conference on Education and research in Computer Aided Architectural Design in Europe, Prague, 355-364.

Turner, A. (2007) 'From axial to road-centre lines: a new representation for space syntax and a new model of route choice for transport network analysis', *Environment and Planning B: Planning and Design*, vol. 34, no. 3, p. 539 – 555.

Wolff-Plottegg, M. (1996) *Architektur-Algorithmen*, Wien: Passagen Verlag.

[1] <http://fold.it>

[2] <http://wp10478398.server-he.de/LabyrinthRunner>

