Bauhaus-Universität Weimar

INFORMATIK IN DER ARCHITEKTUR | INFAR

ARBEITSPAPIERE
WORKING PAPERS

NR. 3, FEBRUAR 2011

GENERATING URBAN STRUCTURES

REINHARD KOENIG

ISSN 2191-2416

Bauhaus-Universität Weimar, Professur Informatik in der Architektur, Belvederer Allee 1, 99421 Weimar
Fon: +49/3643/584201, caad@architektur.uni-weimar.de, http://infar.architektur.uni-weimar.de

Reinhard Koenig
Generating urban structures: A method for urban planning supported by multi-agent systems and cellular automata

Weimar 2011
Arbeitspapiere Informatik in der Architektur, Bauhaus Universität Weimar, Nr. 3
ISSN 2191-2416

Bauhaus-Universität Weimar, Professur Informatik in der Architektur
Belvederer Allee 1, 99421 Weimar
http://infar.architektur.uni-weimar.de

Redaktionelle Anmerkung:

Dr. Reinhard König ist Vertretungsprofessor der Professur Informatik in der Architektur an der Bauhaus-Universität Weimar.

Der vorliegende Text beruht auf der Arbeit:
Koenig, R., & Bauriedel, C. (2004). Computergenerierte Stadtstrukturen. München: Eigenverlag.
http://www.entwurfsforschung.de/compStadt/compStadt.htm

Generating urban structures:

A method for urban planning supported by multi-agent systems and cellular automata

Reinhard Koenig
reinhard.koenig@uni-weimar.de
Professur Informatik in der Architektur
Fakultät Architektur, Bauhaus-Universität Weimar, Belvederer Allee 1, 99421 Weimar, Germany

Abstract

This work is based on the concept that the structure of a city can be defined by six basic urban patterns. To enable more complex urban planning as a long-term objective I have developed a simulation method for generating these basic patterns and for combining them to form various structures. The generative process starts with the two-dimensional organisation of streets followed by the parceling of the remaining areas. An agent-based diffusion-contact model is the basis of these first two steps. Then, with the help of cellular automata, the sites for building on are defined and a three-dimensional building structure is derived. I illustrate the proposed method by showing how it can be applied to generate possible structures for an urban area in the city of Munich.

Keywords: multi-agent systems, cellular automata, generative design system, diffusion limited aggregation, diffusion-contact model, urban modeling.

1. Introduction

Most of the urban structure models published are based on a scale that does not reach down to the scale of architecture and architectural spaces. The majority of models take an economist's or geographer's point of view and see urban structures as the result of complex interactions of individual, ecological and economic relationships.

As one of the fundamental and most important contributions to a generative theory of architectural space we can consider Bill Hillier's alpha syntax model of space (Hillier et al, 1976), further developed in "The social logic of space" (Hillier and Hanson, 1984, page 52-81), where the morphological results of a set of simple rules (syntaxes) are explored to derive simple grammars of form for developing agglomerations of buildings. Other relevant works are "The geometry of environment" (March and Steadman, 1971) where effects of geometry on spatial configurations were examined. On the scale of individual buildings we have to mention the work of Mitchell (1990) as well as Stiny and Gips' (1971) explorations of possible combinations of architectonic elements, summarised under the designation "Shape Grammar".

Inspired by the research mentioned above and the work of Watanabe (2002; 2003), Coates et al (1996), Erickson and Lloyd-Jones (1997) and the research of the collaborative research centre SFB 230 (Teichmann and Wilke, 1996), we have developed a method for combining the generative approach with self-organising principles. In particular, Eda Schaur's (1992) contribution regarding non-planned settlements provides many pointers for describing bottom-up rules, e.g. how elements organise themselves to form more complex structures. As far as possible we have tried to use the bottom-up approach, but not all urban development processes can be described with this method, which is why we have introduced restrictions on different levels to control the system globally using top-down techniques. A flexible combination of both, bottom-up and top-down methods allows the simulation of a wide range of urban patterns and development processes under different conditions.

A central question of our research was to examine how particular structure formations arise in cities. For this we needed to identify the basic types of urban patterns, and here we have drawn on Klaus Humpert's (1992; 1997) concept of six "Feldtypen" (field types): "Nukleus", "Cluster", "Wegelagerer", "Ausleger", "Vernetzer", and "Plan" (figure 1). Adapted from Humpert the complete spectrum of city structures can be produced by combining these basic field types.

A further important task is to define those elements to which the urban phenomena can be reasonably reduced with the intention of developing algorithms for reproducing these phenomena. Linked to this are the respective combination rules of elements required in order to describe the urban processes for producing different structural qualities. We organize the generating process on four thematic levels to develop several strategies for the respective requirements. This allows us to explore the connections between the elements and the rules of their interaction.

The intention of this work is to examine how the principles that underlie existing urban structures can be transformed into algorithms and mathematical parameters to achieve a new comprehension of urban development. The results provide an experimental basis for generating various structures and designs that exhibit different attributes and characteristics. Furthermore the system we have developed allows us to produce alternative solutions for the same design problem very quickly.

The technical basis for the experiments and examinations is the programming (or scripting) language Visual Basic for Applications (VBA). The application used was AutoCAD 2005 which provides a CAD environment in which the generated results can be easily visualised and uses a standard vector format that can be used for further purposes without data conversion problems. The disadvantages are the slow execution of the simulations and the difficulty of implementing an intuitive and interactive user interface within AutoCAD.

2. The six basic urban patterns

Figure 1 shows all six “Feldtypen” (field types) put forward by Humpert. The six types are described at a scale of approximately 1:10.000 and using these basic types it is possible to construct all forms of human settlement. The individuality of a city is not based on the creation of new types, but on the individual arrangement of universal ones. They are compatible in an arbitrary order and mixture ratio. We shall describe each of the individual field types in more detail (Humpert, 1992, pages 88-96).

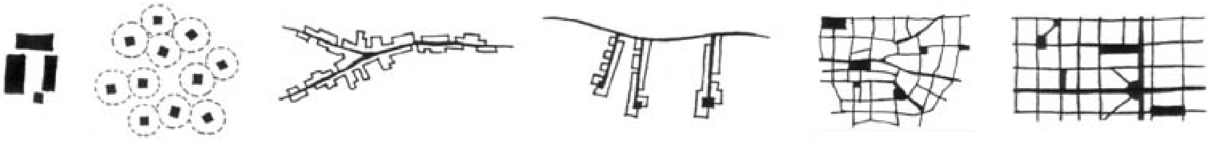

Figure 1: The Six “Feldtypen”. From left to right: “Nukleus”, “Cluster”, “Wegelagerer”, “Ausleger”, “Vernetzer”, “Plan” (Humpert, 1992)

The "Nukleus" (nucleus) – a point type

The nucleus marks the transition from a house to the city. A nucleus is an autonomous and strong building, for example a small group of houses, restricted primarily to a larger plot. In this case, a typical site development pattern of houses along a street is not exhibited. For groups of buildings an inner courtyard assumes this function. A nucleus often marks the initialisation of later settlement activity, but it can also prevent such activities.

The "Cluster" (cluster) – a stochastic field type

In this type, none of the three basic elements, site development, parcelling, and buildings, is fixed from the beginning. A more or less random initial distribution of individual buildings determines the further steps. The development process of a cluster structure can be described in three phases:

(a) Every building claims an "area" around itself, gradually pushing back the public space until it eventually consolidating to a plot; (b) this process of replacement results in a site development system reduced to a linear structure. (c) In the end the site development system can not be reduced any further without collapsing the complete system and it has a much higher resistance than the adjacent parcels. This process of emergence ultimately leads to a logical system of site development, of division of lots, and buildings. Examples of this kind of development pattern are common in many old towns with twisty streets and in the spontaneous, unplanned settlements on the edge of Third World cities.

The „Wegelagerer"(highwayman) – a passive linear type

This type embodies a very economical form of settlement. Sites are built up adjacent to main and minor roads without a development plan. Through the successive addition of one house beside the other a linear building structure arises. Because there are no development costs at the beginning, this type is very economical, but this can change if later improvements to the road networks or extension of the settlement into the hinterland are prevented by the existing structure.

The "Ausleger" (boom) – an active linear type

In contrast to the "Wegelagerer" (highwayman) where the road already exists, for this type a public or private road has to be built especially. Normally this new street is more or less perpendicular to an existing main road and utilises a stock of sites. This type can be newly-created on a single large plot of land or it can make use of old paths.

The „Vernetzer"(interlink) – an interactive field type

This type is usually created on the basis of older paths or overland routes that are iteratively regulated step by step, or through the insertion of single new streets to interlink old structures. Some forks, "slopes" and crooked routes which can be seen most clearly in overland roads are gradually eliminated, as the increasingly dense packing of plots eventually forces their own proportion back on the site development network.

The gradual transformation to a more orthogonal ground plan for the settlement structure is not at all the result of a planning intention. The right angle results from the dynamics inherent to systems of growing settlements. Since this process never comes to an end, the system of site development still exhibits small disturbances in many locations. The "Vernetzer" is the direct transformation of farmland and grazing land into building land.

The „Plan"(plan) – a deterministic field type

For this type man appears as a planner. He defines the site development, the parcelling plan, and the structure of the buildings. In extreme cases this determination extends as far as the architectural design of each building. If the planning specification confines itself to site development only the future settlement can be purely stochastic, i.e. carried out in an individual style. But every plan runs the risk that it will be never put into practice completely and will remain simply a limbless body.

3. The structure of the simulation model

To facilitate the clear processing of the complex task of generating urban structures with the help of simulation models, four levels are introduced as sub-models to deal with general information, site development, buildings, and optimisation. The information level can be considered as a dynamic database for storing and retrieving local information. Our main activities concentrate on the development of the following two levels. Based on a contact model, the concept of the development level is to examine possibilities for generating different "Feldtypen" using road networks. For the building level, cellular automata are used to design two and three-dimensional building structures. At the optimisation level, methods for measuring and analysing the generated structures are presented and a conceptual draft for optimising structures is mentioned. On all levels the abstract simulation world consists of a rectangular lattice of cells.

3.1. Information Level

The information level is firstly a means of processing information that cannot be reduced to basic elements. For example environmental information about vegetation, topography, conditions of the soil and the weather as well as statistical information about population, demography, crime, and social structure can be stored as aggregated data in the cells of the lattice. Secondly, this level can be considered as an interface to geographic and economic models on a larger scale. In principle this level is equivalent in function to a loose coupling with a Geographic Information System (GIS) (Wegener, 2005). The models developed at other levels can access the database to adapt their steering parameters and manipulate the stored values.

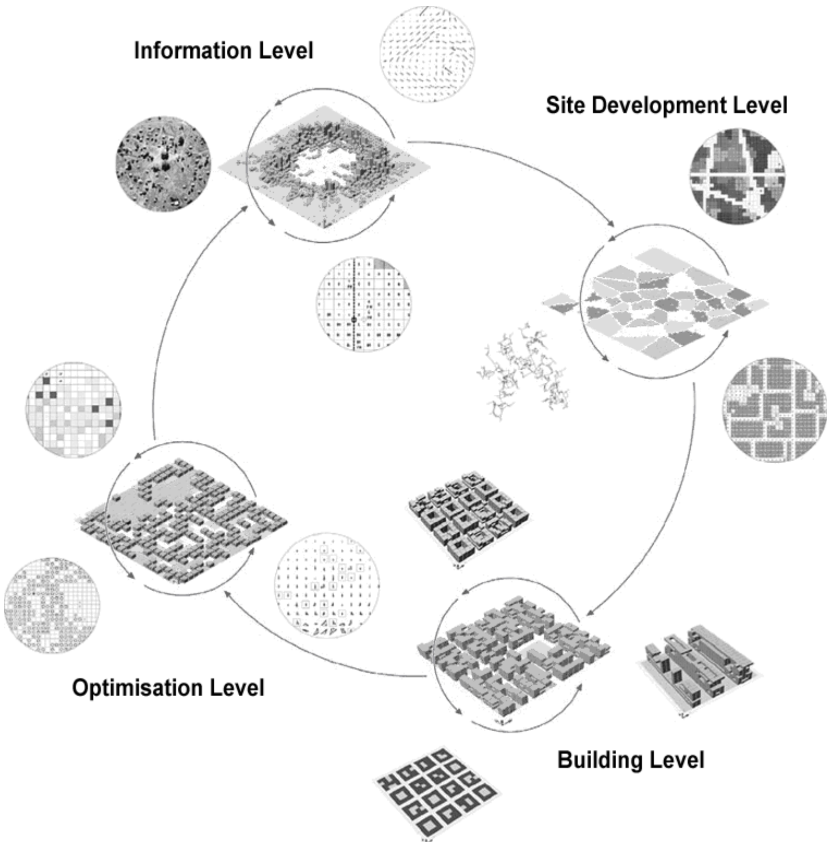

Figure 2: The four basic levels of the simulation model.

3.2. Site Development Level

'Streets are major and houses are minor' – this old saying seems to be true even today. (Watanabe, 2002)

The organisation of the road network is one of the main structuring principles of a city, closely related to the subdivision of the urban area into individual parcels. By controlling the simulation of an agglomeration process such as diffusion limited aggregation (DLA)

(Flake, 1998, pages 71-75), structures with different characteristics can be generated and identified as the aforementioned "Feldtypen". All kinds of urban structures should be realisable by combining these types.

In his Induction Cities project, Watanabe (2002) describes the positive effect of a maze of winding paths in the old parts of towns in contrast to settlements in a raster layout with streets running in a straight line. In contrast to Watanabe, in our assessment we want to go beyond a comparison of what are beautiful and what are functional streets. For us a method for comparing the total length of a road network with the necessary length of detours required to reach a destination in such a system seems more interesting (Schaur, 1992). The structure of a site development system is directly related to its land and energy consumption for its construction or the use of the system.

3.2.1. Diffusion-contact model

Our examinations at this level begin with a diffusion limited aggregation (DLA) process which can be denoted as contact model (Batty, 2005, pages 47-51). In DLA processes particles follow random paths in Brownian motion until they cluster together to form aggregated structures (Witten and Sander, 1981). DLA is applicable in any system where diffusion is the primary means of transport in the system. It can be observed in systems such as electrode position, Hele-Shaw flow, mineral deposits, and dielectric breakdown, where a structure gradually grows over the course of time by adding particles to an existing structure.

To transfer this process to a computer simulation, a two dimensional regular lattice of squares is used as a Cellular Automata (CA) (Toffoli and Margolus, 1987). At a Cellular Automata Z each one of the i cells of the lattice gets assigned an index $H = \{1, 2,\ldots, i\}$ and can be in one of k possible states $S^H = \{S_1, S_2,\ldots, S_k\}$. For the simulation of a DLA at least three states k=3 have to be defined. First, for empty cells the state $S^H_1 = 0$, second for occupied cells where a particle is aggregated the state $S^H_2 = 2$, and third for cells where an aggregation is possible the state $S^H_3 = 1$ (candidate sites). Further we need to represent the particles in the simulation. For this purpose Brownian agents (Schweitzer, 2003) are introduced. These m agents $A = \{A_1, A_2,\ldots, A_m\}$ can move freely across the cellular space and interact with cells located at the same position. Normally it is useful to restrict the movement of the agents on a CA to discrete steps from cell to cell. Following Portugali (2000) the complete system of CA and interacting free agents can be denoted as "Free Agents in a Cellular Space" (FACS). Lastly, for a DLA simulation a transformation rule F for the cells is necessary.

F can depend either on the states of the cells in the neighbourhood $U(H)$ of cell H, thus $S^{U(H)}$ denotes the set of states of the neighbouring cells, or F depends on the position P^A of agent A. Now, we can write down the general transition rule of our FACS model for a simple DLA process:

$$S^H_{(t+1)} = F_t\,(S^H(t),\ S^{U(H)}(t),\ P^A(t)). \qquad (3.1)$$

In summary this means that the state S of cell H in the next discrete time step $t+1$ depends on the transition rule F at time t. The transition rule includes the state S of cell H at time t, the neighbourhood's configuration S^U of cell H at t and the position P of agent A at t. In the same way the movement rule L for the position of agents A can be written:

$$P^A(t+1) = L_t(S^H(t),\ P^A(t),\ S^{U(H)}(t),\ P^{U(H)}(t)), \qquad (3.2)$$

where $P^{U(H)}(t)$ restricts the possible locations where an agent A can move to randomly at the next time step. The states S of the cells H indicate if the agent can aggregate at a location, or if the agent is allowed to enter a certain cell at the next time step.

With this formal equipment we can operationalise the model's behaviour in more detail. To start the process we have to place at least one agent at a random chosen location H_b at the border of the CA lattice and define at least one cell as aggregated. For this first example we take the cell H_c in the center of the field as the seed (the light grey cell in figure 3, left) and all other cells i are assumed to be empty. The initial conditions for the most basic process of DLA can now be written as

$$P^A(0) = random(H_b),\ S^{Hc}(0) = 2,\ S^{Hi}(0) = 0,\ \forall i \neq c\ . \qquad (3.3)$$

In the next time step, the transition rules of the CA and the agent(s) are executed. We need to first define the neighbourhood $U(H)$. Here we have chosen the Moore neighbourhood containing the eight surrounding cells of the cell in question (the eight dark grey cells in figure 3, left). A cell changes its state from 0 to 1 under the following condition:

$$S^H(t+1) = 1,\ \ if\ \ (S^H(t)=0\ \ and\ \ C^{U(H)}(t)>0\ ), \qquad (3.4)$$

where

$$C^{U(H)}(t) = \sum_G \left\{ 1 \mid G \in U(H),\ S^G = 2 \right\} \qquad (3.5)$$

is the counter of the crystallised cells with state $S=2$ in the neighbourhood of cell H. Consequently an aggregation is possible only where at least one cell in the neighbourhood has already been crystallised. To change a cell's state from *1* to *2* a further connection condition is defined by:

$$S^H(t+1) = 2,\ if\ (S^H(t) = 1\ \ and\ \ P^A(t) = H(t)). \tag{3.6}$$

If condition (3.6) is met, the current agent is deleted and a new one is created at *random* (H_b). After the execution of the CA the agent(s) moves one random step across the cellular field:

$$P^A(t+1) = random(P^{U(H)}(t)), \tag{3.7}$$

where the position of an agent is selected randomly from the adjacent cells into which it is allowed to move in one step. In this example the same eight Moore neighbourhood cells *U(H)* are taken for $P^{U(H)}$ as for the CA. Figure 3 shows the aggregation of an agent after a random walk (on the left) at the point where it enters the neighbourhood of an already crystallised cell, and the resulting change of the cell's state from *1* to *2* (on the right).

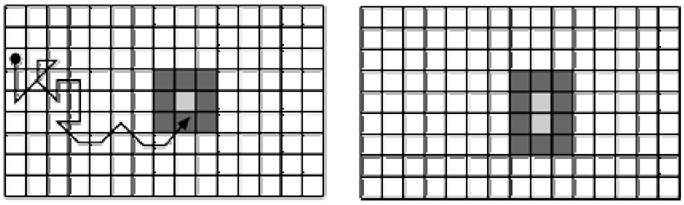

Figure 3: The contact rules for the diffusion limited aggregation (DLA) process.

An example of the result of the DLA process described can be seen in figure 4a. A useful adaptation of the connection condition (3.6) is to introduce a probability ρ for the connection of a particle:

$$S^H(t+1) = 2,\ if\ (S^H(t) = 1\ \ and\ \ P^A(t) = H(t)\ \ and\ \ \varepsilon < \rho), \tag{3.8}$$

where ε is a randomly chosen value between 0 and 1. At $\rho=1$ a particle is connected every time if it crosses a cell with state 1. Accordingly the probability is relatively low that a particle will reach the centre of a cluster without connecting beforehand. Because of this, the growth of the cluster occurs mainly at the edge of the structure. If the connection probability ρ decreases, more particles can pass the edge and growth also takes place within the cluster (figure 4b). The structure at figure 4c is generated by placing two initial cells and revokes the restriction that agents can start their random walk at the border of the cellular field only. In this case two growing clusters can merge into one structure.

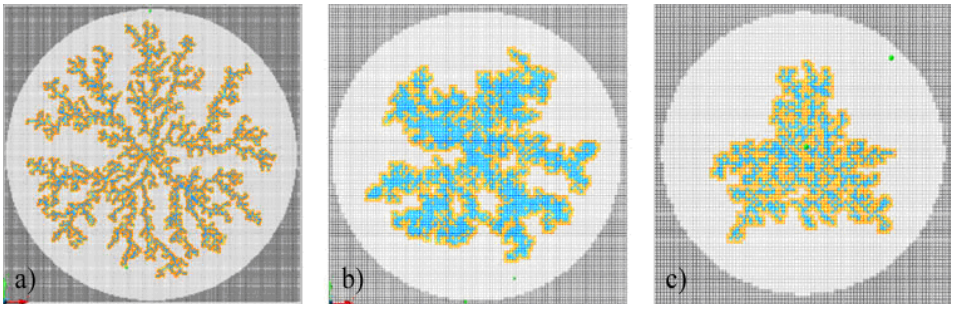

Figure 4: Three resulting structures of the DLA process: a) a field with 200 x 200 cells, $\rho=1$ and 3127 connected elements; b) a field with 100 x 100 cells, $\rho=0,1$ and 1650 connected elements; c) a field with 100 x 100 cells, $\rho=0,5$, the starting location of an agent $P^A(0)$ is not restricted to the cells H_b at the border and all random walks can start at any cell H_i.

3.2.2. Generative methods for various road networks with the help of the field type concept

How can we progress from the abstract DLA process to a growing network of streets and roads? To answer this question we start at the scale of the city as a whole. It is possible to describe the approximate development of a city over time using a CA model and a process adapted from the DLA (Batty and Longley, 1994). Using a similar approach we undertake a first attempt and begin with an empty landscape in which a system of roads begins to grow step by step. To transfer the DLA structure to a network of roads, the crystallised cells are considered as nodes *N* (crossing) and are connected by an edge *E* (street) of length *d* to other nodes that are located inside a circle with radius *r*. The connecting rule is illustrated in figure 5a. The process starts with $N0 = S^{Hc}(0) = 2$ and the nodes *N1* until *N4* are connected one after the other by drawing an edge to *N0* only if d > *r*.

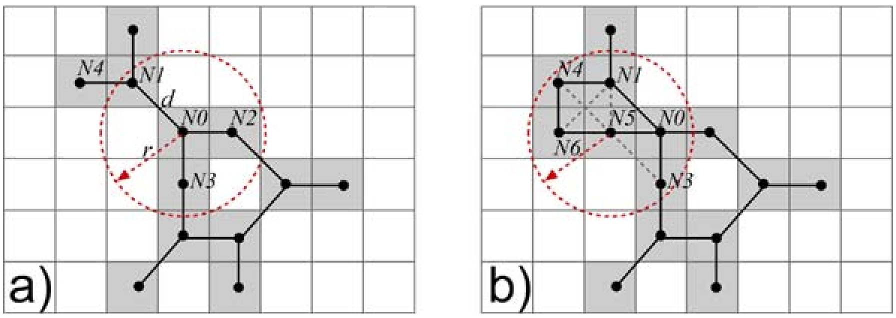

Figure 5: The rules: how to build a network of roads on the basis of a DLA process.

Using this method we can obtain structures such as those illustrated in figure 6. The problem, clearly evident in the picture on the right of figure 6, is the high density of connecting edges in the road network for relatively small values of ρ. To solve this problem another restriction has to be introduced. If a new node *N5* (figure 5b) is connected to a current cluster (with *N0 – N4* and still without *N6*), an edge is drawn to one of the nearest nodes (*N0*) and then the shortest path is calculated to the other connection candidates. The nodes that one has to pass through to reach a connection candidate via the current graph are counted in *C*. To compute the shortest paths the A* algorithm is used (Russel and Norvig, 2002). To discuss this algorithm in greater detail would go beyond the scope of this article. Now we can write the condition to draw an edge *E* as:

$$E_{(Ni,\, Nj)} \; If\,(d < r \;\; and \;\; C > X), \tag{3.9}$$

where *X* is the minimum of required nodes between a new node and a connection candidate before a new edge is drawn. In the example in figure 5b *X=3*. By manipulating the three main parameters *U(H)*, the neighbourhood of *H* where a further connection is possible, and the probability ρ to regulate the density of the growing cluster, as well as *X* to control the frequency of the connections, we can generate global road networks with various characteristics. In general, the possible road networks can be classified in three categories. First, a tree structure with dead-end streets, second a network where each node is connected at least two other nodes, and third a mixture of the first two categories where some branches are dead ends, which can be denoted as a semi-lattice (Alexander, 1965).

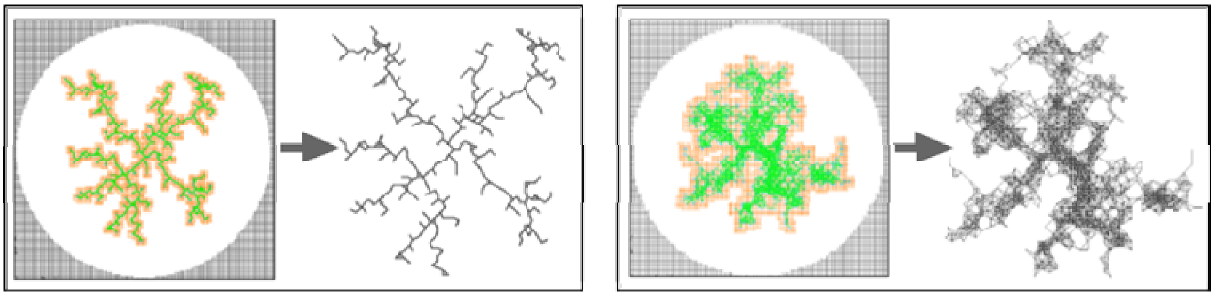

Figure 6: Possible road networks without restriction for the connections.

Now, if we change the scale from the city as a whole to a district or neighbourhood and consider a basic road network as pre-determined either from the generated network above or from existing older paths or overland routes, we can work out the generative methods of the field types in more detail. The first type to consider is the simplest one, the "Wegelagerer". To generate this type we just need to consider the pre-determined roads as crystallised seeds which different agents can connect to using the connection condition (3.6). After connecting, the agents occupy a plot of land (figure 7). For the occupation process there are various possibilities. For example, the agents can keep a site of a given size. This method is similar to a kind of Tetris game (figure 7b). Alternatively, after all agents have been connected they can expand the plot they occupy by spreading to the neighbouring cells step by step until they adjoin other expanding plots (figure 7a). The areas with the same colour can be considered as "Nucleus" types. The widening process is described in the text below (3.14 – 3.17).

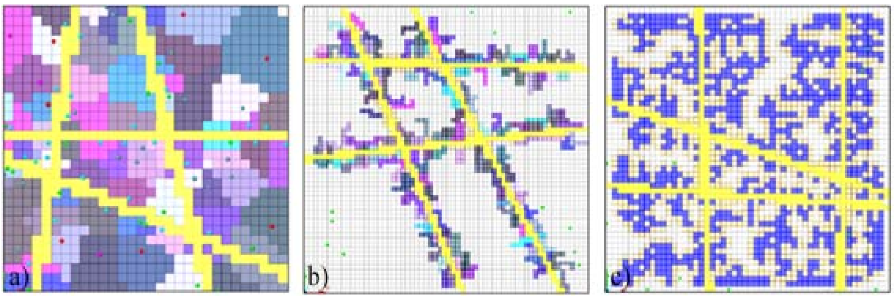

Figure 7: Generated "Wegelagerer": The agents connect to pre-determined streets and occupy a plot of land.

However, if the agents are allowed to build in the hinterland, the site development is missing (figure 7c). The "Ausleger" is the obvious type for generating secondary site development. Two different methods are developed for this type. The logic of the first is similar to the "Wegelagerer" system, but in this case the agents can not only connect to streets as crystallised seeds but also to the neighbourhood of plots of land already occupied by aggregated agents. As before the connection condition (3.6) is applicable for this type. After an agent is aggregated and has occupied a site, the site development is constructed by drawing the shortest path to an existing street, again using the A* algorithm. Figure 8 shows the resulting structures with different plot sizes (figure 8a and 8b) and with all plots coloured blue (figure 8c).

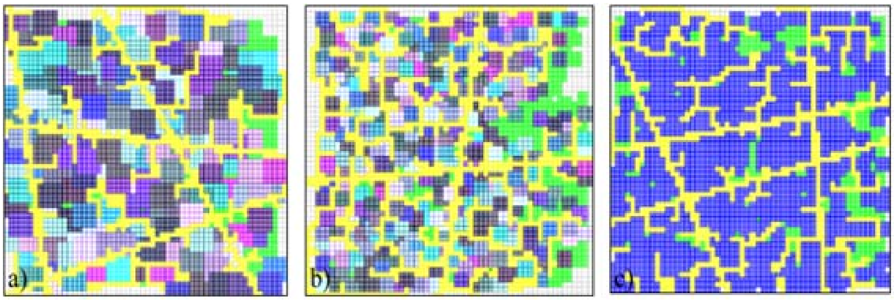

Figure 8: Generated "Ausleger". Method 1: The new aggregated sites are developed by new streets and are thereby connected to the next road.

In order to find a way of moving on to the "Vernetzer" type, we introduce a second method of generating "Ausleger" types. In this case, the generative logic is inversed: aggregated agents no longer represent a site, but the extension of a street flanked by plots of land on both sides. In addition to the three cell states previously described, empty, occupied (street), and candidate sites for possible aggregation, a fourth state $S^H_4 = 3$ for a plot of land is added. The condition for a cell to change its state to "plot of land" is similar to (3.4):

$$S^H(t+1) = 3, \quad if \ (S^H(t)=0 \ and \ L^{U(H)}(t)= 3), \qquad (3.10)$$

where

$$L^{U(H)}(t) = \sum_G \{1 \mid G \in U(H),\ S^G = 2\} \qquad (3.11)$$

is the counter of the street cells with state $S=2$ in the eight Moore neighbourhood cells of the cell under consideration H. The order of execution of the different rules per time step is as follows: first check if an agent can connect (3.8), second see if the cells can be converted to plots of land (3.10), and third find the cells where further aggregation is possible. For the last rule the counting condition (3.5) has to be modified to

$$C^{U(H)}(t) = \sum_G \{1 \mid G \in U(H),\ S^G = (2,\ 3)\}\ . \qquad (3.12)$$

This alteration means that not only streets are counted but also plots of land. As a last rule for a time step, the probability ρ for the connection of an agent at equation (3.8) has to be computed. Now, the probability ρ is defined locally for each cell and depends on the states of the neighbouring cells:

$$\begin{aligned}
&\rho^H(t) = 0{,}1 \quad \textit{if} \qquad L^{U(H)} = 3 \qquad \textit{otherwise} \\
&\rho^H(t) = 1 \quad \textit{if} \quad L^{U(H)} < 3 \qquad \textit{otherwise} \qquad (3.13) \\
&\rho^H(t) = 0; \\
&L^{U(H)}(t) = \sum_G \{1 \mid G \in U(H),\ S^G = 2\}\ .
\end{aligned}$$

These basic rules can be adapted to widen the area of the plots of land next to the roads. As a result, empty cells are considered and are assigned the state of the cells in the von Neumann neighbourhood when they are not empty. The von Neumann neighbourhood consists of the four adjacent cells in the north, east, south, and west directions. Finally those empty cells which have a cell with the state "plot of land" in their von Neumann neighbourhood can defined as candidates $S^H = 1$ and are assigned the probability $\rho^H = 0{,}1$ (figure 9a – c).

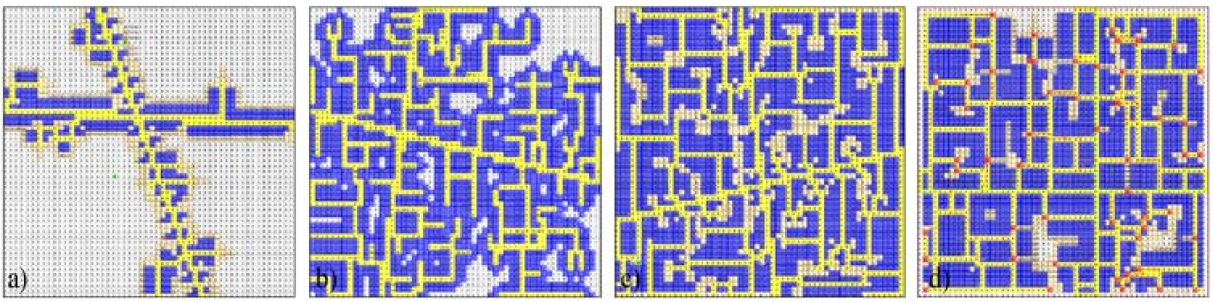

Figure 9: a) – c) Generated "Ausleger" with method 2. New streets are generated by aggregation and are flanked by plots of land on both sides d) Generated "Vernetzer". The open nodes are marked red and are connected by a street if they coincide near enough.

After all the previous modifications, it is now easy to derive the generative method for the field type "Vernetzer" from that of the "Ausleger" type. For this a new rule has to be added at the end of the calculations per time step. If a street cell has an empty or a candidate neighbour cell in its von Neumann neighbourhood it can be considered as an open node and we can look for other open nodes in a certain distance (analogous to the case in figure 5, $r > d$). Two open nodes are connected by a new street if condition (3.9) is met. To identify the steps to the other open node under consideration, the street cells that one has to pass through to get from one node to the other are counted in C. The parameter X gives the minimum of required street cells between two open nodes before a new street is built (figure 9d).

Finally we will also examine the "Cluster" type. For this the same principle as for the DLA process (3.3 – 3.8) is used for spatial arrangement. The only modification to the connection condition (3.6) is that the Moore neighbourhood under consideration $U(H)$ is enlarged to radius $r=5$. This means that the neighbourhood now consists of $11 \times 11 = 121$ cells (figure 10). After all agents have connected, they first use their individual agent index m to stamp a mark $M^H=m$ on the occupied cell. Afterwards the occupied plots $(S^H=3)$ are widened to the neighbouring cells step by step through diffusion until they adjoin with other widening plots:

$$S^H(t+1) = 3, \quad if \quad (S^H(t)=(0 \text{ or } 1) \text{ and } L^{U(H)}(t) > 0), \qquad (3.14)$$

where

$$L^{U(H)}(t) = \sum_G \left\{ 1 \mid G \in U(H),\ S^G = 3 \right\}. \qquad (3.15)$$

The mark M^H of the new plot cells is given by the average of their neighbouring plot cells:

$$M^H(t+1) = \frac{C^{U(H)}(t)}{L^{U(H)}(t)}, \qquad (3.16)$$

where

$$C^{U(H)}(t) = \sum_{G \in U(H)} M^G. \qquad (3.17)$$

The street cells are generated between different plot cells as indicated by their different marks M^H. The condition for a cell to change its state to a street can be written as:

$$S^H(t+1) = 2, \quad if \quad \frac{C^{U(H)}(t)}{L^{U(H)}(t)} \neq M^H, \qquad (3.18)$$

where $C^{U(H)}$ is taken from (3.17) and $L^{U(H)}$ from (3.15). This change is independent of the state $S(t)$ of the current cell.

The last field type "Plan" is not discussed here because the structure of streets and sites for this type is determined by a human planner. In the next section we use a raster grid as an example of a "Plan" type, where the positions of the streets are defined by the distance between them.

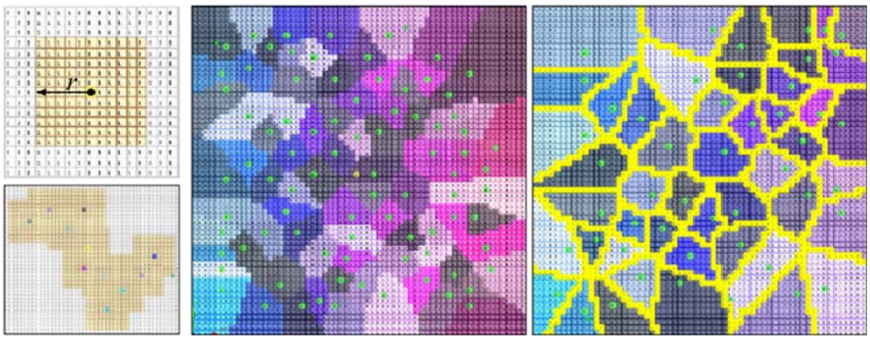

Figure 10: Generated "Cluster". The sites are placed by a DLA process and the streets are built between the sites.

3.3. Building Level

In the framework of this level we concentrate on the question of how the sites in a given road network can be built on to meet certain conditions. For this purpose a first step explores two-dimensional structures where cells are distinguished between the categories empty or built on. The two-dimensional structures form the basis for the generative methods of the three-dimensional building structures in the next step. Finally we briefly discuss some experimentation with the growth of "free" three-dimensional structures.

3.3.1. Two-dimensional structures

At the building level only Cellular Automata are applied and the first task is to find appropriate rules for binary automata to produce results usable to fill an area with buildings. To examine the automata a formal description for counting rules and voting rules are introduced. The voting rule means the kind of neighbourhood denoted above with *U(H)*. In the following, if not otherwise stated, the eight cells Moore neighbourhood is used for *U(H)*. The counting rule describes the condition for a cell to change its state. At binary automata there are only two states S^H=0 for an empty cell and S^H=1 for a built on cell. The explicit formulation of the counting rule is as follows:

$$S^H(t+1) = 0, \; if \; (C^{U(H)}(t) >= B_1 \;\; or \;\; C^{U(H)}(t) <= B_2),$$

$$S^H(t+1) = 1, \; if \; (C^{U(H)}(t) > B_3 \;\; and \;\; C^{U(H)}(t) <= B_4), \qquad (3.19$$

$$S^H(t+1) = S^H(t), \; else.$$

where

$$C^{U(H)}(t) = \sum_{G \in U(H)} S^G \, , \qquad (3.20)$$

and B_n={ B_1, B_2, B_3, B_4} denotes the particular thresholds for a cell, if its environment is too densely occupied and the cell has to change to an empty state, or if there is potential and demand in the neighbourhood to expand the built on states and change the cell's state to 1. To abbreviate the spelling of the global counting rule *R* of CA the thresholds B_n can be summarised to $R=B_1B_2B_3B_4$. Using this formal equipment the exploration of automata which generate structures where each built on cell has at least one empty cell in its von Neumann neighbourhood can be documented. This restriction seems sensible to ensure enough light and air for each house. The initial state of an automaton is normally given by

$$S^H(0) = random\,(0, 1), \qquad (3.21)$$

with the probability $\rho^H(0)=$ 0,5 or 50% that at the outset a cell has either the state 0 or 1. The first automata under consideration has the rule $R_1=8114$ and produces structures with a high density D=0.72 of built on cells. The density *D* is a measure of the ratio of built on cells to all cells. The problem with the structures resulting from rule R_1 is that there are quite a lot of built on cells without a free cell in their von Neumann neighbourhood (figure 11a). By altering the rule R_1 to $R_2=7013$ this difficulty is removed for nearly all cells as illustrated in figure 11b though the density is lowered to D=0.61. If the requirement for empty cells in the von Neumann neighbourhood of each cell rises to at least two empty cells, the rule $R_3=5012$ is suitable if a further lowering of the density to D=0.46 is acceptable or desirable. To produce structures with an even lower density the rule $R_4=3011$ can be used resulting in D=0.37 (figure 11c).

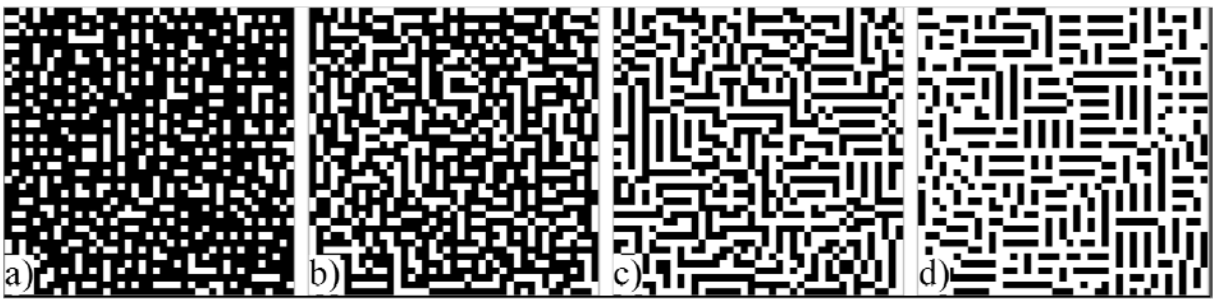

Figure 11: The illustrations show four different possibilities for filling an area with buildings, where the white cells are the empty ones with $S^H=0$. The generative rules *R* and the densities *D* for the particular structures are: a) R_1=8114, D=0.72; b) R_2=7013, D=0.61; c) R_3=5012, D=0.46; d) R_4=3011, D=0.37.

If more rural structures are required the rules R_5=4010 and R_6=3010 are appropriate to generate structures with D= ~0.21 and D= ~0.10, but to achieve suitable results the initial probability has to be changed to $\rho^H(0)$= 0.3. In the acknowledgement at the end of this article the link to an online simulation tool is provided to enable interested readers to examine the automata rules and dynamics for themselves.

For the next elaboration of the settlement structure two new states for the cells are introduced. First, for the streets marked in yellow the state S^H=2 is reserved and second, if in case during the generative process a 3 x 3 cells area without built on cells results by chance, theses cells are kept free during the further development by placing a red coloured cell with the state S^H=3 in the midpoint. The counting rule remains the same as given in (3.19) for all cells with state S^H=(0, 1), but (3.20) have to be changed to:

$$C^{U(H)}(t) = \sum_G \{1 \mid G \in U(H),\ S^G = 1\}\ . \qquad (3.22)$$

At setup, first the street cells are defined and afterwards the initial states of the remaining cells are set to:

$$S^H(0) = random\ (0, 1),\ \ if\ \ S^H(0) \neq 2. \qquad (3.23)$$

Once a cell has been set to a street state it cannot change its state any more. For the further examinations the rules R_2 and R_3 are considered as most usable and are applied to a pre-determined grid of streets shown in the examples in figure 12. A main problem of the rectangular cell structure of the cellular automata used, beside the restrictions of the geometry itself, is to choose a reasonable size for the cells that is suitable at the same time for the streets, the buildings and the necessary space between the houses. Erickson and Lloyd-Jones (1997) have presented a model with a more irregular combination of rectangular cells but the cell sizes are also the same for all elements.

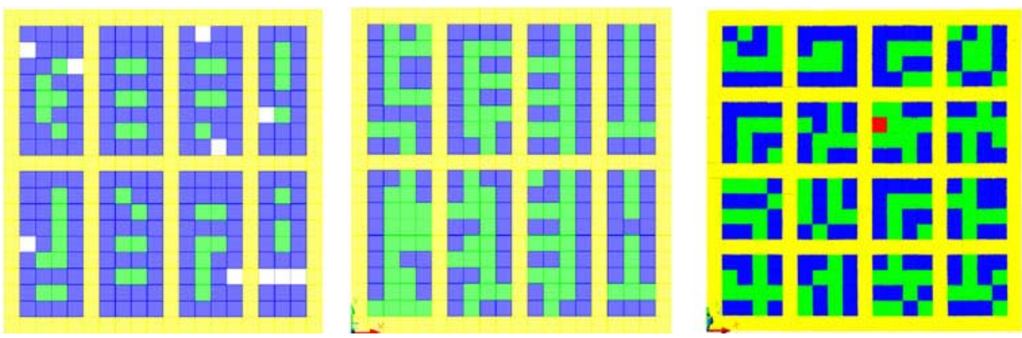

Figure 12: Building structures shown in blue inside a preset grid of yellow coloured streets, the empty or free cells are shown in white or green. a) A block structure generated with R_2=7013; b) A structures with a lower density produced with R_3=5012; c) R_3 applied to a more fine-meshed grid of streets.

3.3.2. Three-dimensional structures

The two dimensional structures of streets, built-on cells and free cells are the basis for the generative methods of three dimensional buildings. Therefore we need to expand the logic of the two-dimensional cellular automata to the third dimension. The counting rule can remain the same as (3.19), however, the considered neighbourhood has to be changed in the way that the cells at the floors above and below the current one are additionally included as shown in the right hand image of figure 13. Thus in the third dimension the Moore neighbourhood contains 26 cells and the von Neumann neighbourhood 6 cells. The 3D automata are binary ones with the states S^H=0 for empty cells and S^H=1 for occupied or filled cells.

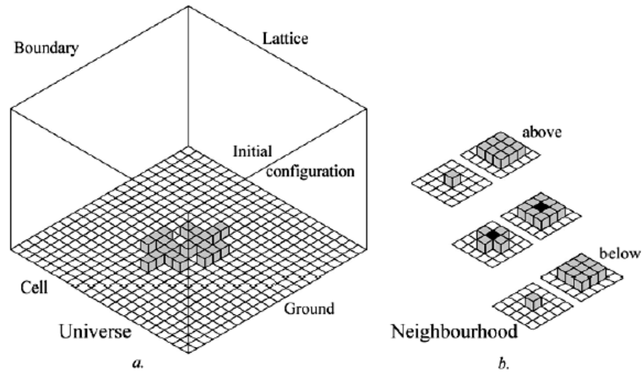

Figure 13: Configuration and terminology for a three-dimensional Cellular Automata (Krawczyk, 2002)

The initial configuration for a 3D automaton is given by transferring the built on cells from the two-dimensional lattice to the ground floor of the three-dimensional lattice by the rule:

$$S^{H0}_{3D} = 1, \; if \;\; S^{H}_{2D} = 1, \text{ otherwise } S^{H0}_{3D} = 0. \qquad (3.24)$$

All cells at the upper floors are in state 0. Because the cells at the ground floor have no neighbouring cells in the floor below these cells are defined as empty. The same is valid for the uppermost floor and the cells above it as well as for the cells at the boundary in the north, east, south and west where the area is surrounded with street cells. The process starts at the first floor and can be bounded to a fixed maximum number of floors. Further restrictions for the 3D automata, for example not to fill cells above streets at all floors, are possible but not necessary. For the initial examination it has turned out that a combination of the Moore (horizontally) and the von Neumann neighbourhood (vertically) is most usable where the eight cells on the same floor and the one cell below the current one are considered as *U(H).* With this voting rule the investigated counting rule to transfer the 2D block structures generated with rule R_2 to an appropriate 3D structure is R_1^{3D}= 6114. The resulting building structures show some variations, but in principle they can be considered as a good and interesting translation of the basic two-dimensional patterns (figure 14, upper row).

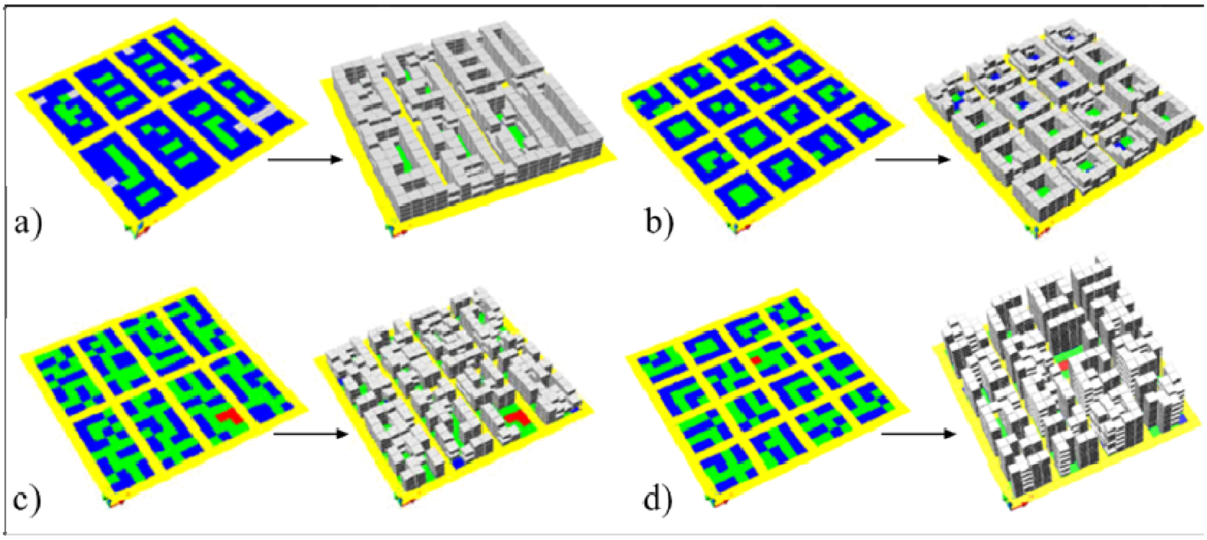

Figure 14: Translation of 2D patterns to 3D building structures. Upper row: The 2D patterns were generated with the rule R_2=7013 and the 3D structures with the rule R_1^{3D}= 6114. Lower row: The 2D patterns were generated with the rule R_3=5012 and the 3D structures with the rule R_2^{3D}= 4011.

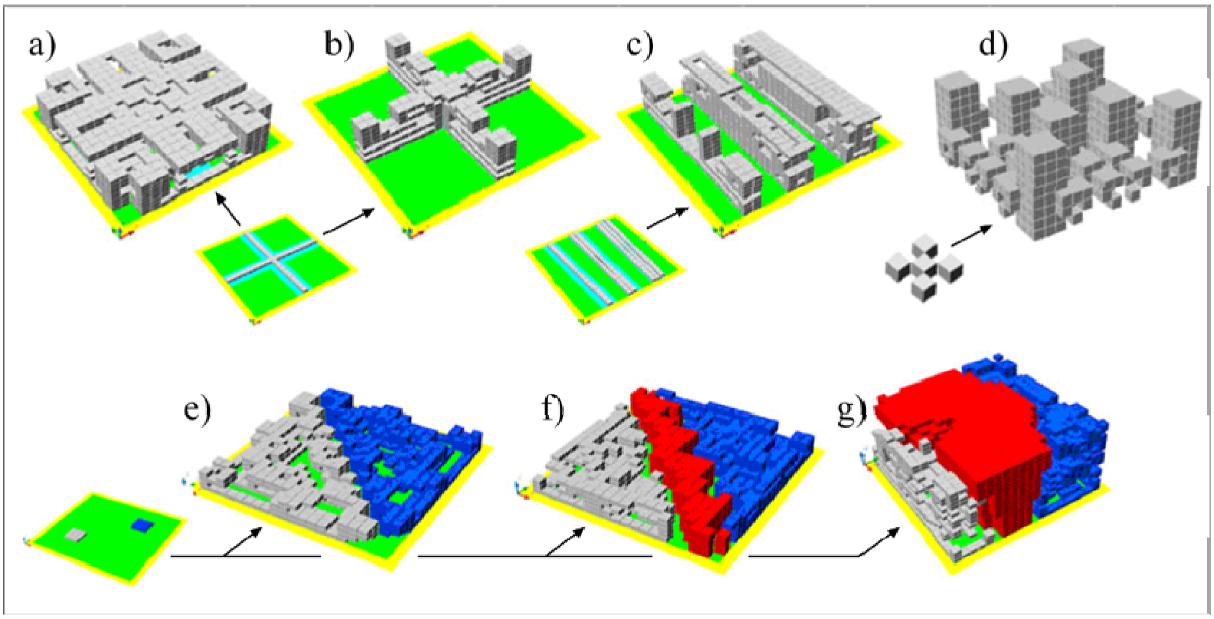

Figure 15: Three dimensional cellular automata (grey and blue) with experimental counting rules can develop freely in space and are only restricted by streets or dynamic cells (red) which have to be kept free and are created if two automata collide. a) R_3^{3D}= 6124; b) R_3^{3D}= 6124, restricted to the magenta coloured cells; c) R_4^{3D}= 6015, after 8 generations; d) R_5^{3D}= 2011; e) R_6^{3D}= 6125; f) grey: R_6^{3D}= 6125, blue: R_4^{3D}= 6015; g) grey: R_6^{3D}= 6125, blue: R_4^{3D}= 6015, red: R_7^{3D}= 9099.

Because it was not possible to find a suitable counting rule to achieve an adequate solution for a translation of the 2D structures generated with rule R_3 to the third dimension, the voting rule was modified to the four cells of the von Neumann neighbourhood at the same floor and the cell directly below the one in consideration. With this new voting rule the counting rule R_2^{3D}= 4011 leads to the expected results illustrated in figure 14, lower row.

The very simplest method to change from a two-dimensional to a three-dimensional structure is just to extrude the built on plots with a defined height, but because there is no difficulty with this implementation it is not investigated in more detail. Finally it has to be noted, that the generated building structures are just rough outlines of possible buildings and they have to be elaborated in more detail, as was done for example in the work of Krawczyk (2002).

To conclude this section on the building level, some further experimental CA are presented succinctly. The development of these automata is highly sensitive to the initial condition of cells in state 1. In figure 15 the small pictures are the initial cell configurations and the large illustrations are the resulting structures produced with different rules and restrictions after a few generations. At the lower row of figure 15 three ways of keeping a certain distance between two automata are shown. The ideas for the freely developing 3D automata are inspired by (Coates et al, 1996).

3.4. Optimisation Level

This level is drafted to deal with concepts for reconstructions of given structures with regard to particular criteria such as land use, light, aeration, rational parcelling, minimising of the site developments, or distribution of usages, et cetera. To optimise a certain settlement structure, the essential variables have to be recorded and evaluated first. The two main variables for a building structure can be determined as the lot coverage and the total floor area. It is relatively simple to compute these values from the cellular automata model by just counting the corresponding cells and calculating the ratios, as done above (figure 11) for the density information D, which corresponds to the lot coverage. The total floor area is the ratio of the occupied (grey) 3D cells to the total area or the sum of all 2D cells (figure 16).

With the methods of the CA developed so far it is only possible to generate structures by pre-determining some elements like streets and the definition of the local counting and voting rules, which has the advantage that a condition is always met locally and consequently globally. But there is no way to feed back the global state to a further development or to state contradictory requirements to the generative system. In the following, two promising methods are introduced: how to make a request that on the one hand cannot be completely met and on the other hand cannot be generated simultaneously by the system. In the scope of this article these two methods are mentioned but not investigated in detail.

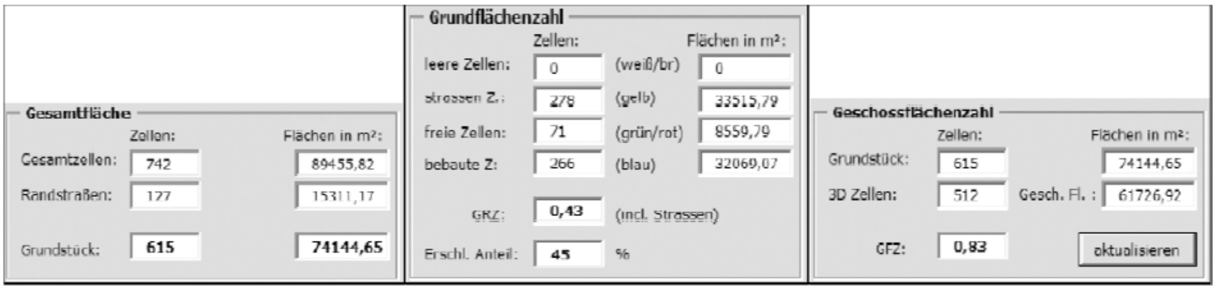

Figure 16: The variables of the total area (Gesamtfläche, left), the lot coverage (Grundflächenzahl GRZ, middle), and the total floor area – (Geschossflächenzahl GFZ, right).

First, a termite system is considered, where termites (=agents) analyse their environment that consists of a defined neighbourhood *U(H)* and make some changes in it (Resnick, 1994; Flake, 1998). For example the density of a certain structure can be increased or decreased at different places after the local condition is compared with the respective requirements. To perform this operation the agents can "take" the built-on state of a cell and "place" it into a cell in a more suitable neighbourhood or delete it if the total number need not to be maintained (figure 17, left).

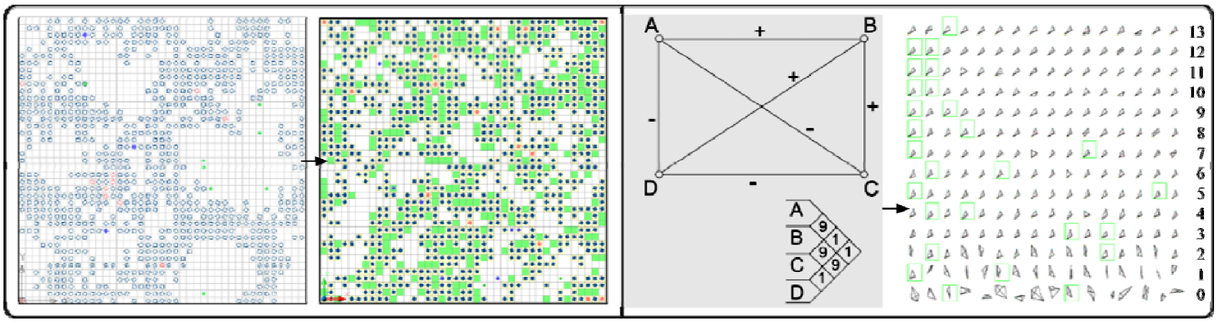

Figure 17: Optimisation methods. Left: A partly highly dense structure is rebuilt and decreased in its local density using a termite system. Right: Distribution of four different usages A, B, C, and D with distance relations (9 = near together or 1 = far away from each other) given in the table. The corresponding figure at the side shows 13 generations of varying arrangements generated by a genetic algorithm.

Second, an interesting task would be to assign a given catalogue of usages to the generated building structures, whereas the distribution of particular usages depends on the allocation of other usages. The conditions can be captured in tabular form where the relative distances between the usages are expressed by numerical proportions. At the definition of these relations some requirements can contradict each other, which is why usually cannot be satisfied all together. Nevertheless an optimal solution with the least possible deviation can be found using genetic algorithms (Goldberg, 1989). The landscape of possible solutions for such problems can include one or more global solutions or there are many more or less well-fitting local optima which can be explored using different techniques. An example of an abstract solution for the relative distribution of four usages is illustrated in figure 17, on the right.

4. Application

In this section, the methods developed above are brought together to show the continuity of the generative planning process. We first briefly review the different structures to illustrate similarities with structures that can be found in reality and afterwards various planning alternatives for a specific urban area in the city of Munich is presented.

4.1. Structural Survey

In the following we consider nine different structures to investigate their characteristic properties and the variations of possible building structures based on the same road network. The attempt has been made to provide a significant spectrum of variants from the multitude of possibilities. Figure 18a shows the initial configuration for the nine squares in 18b, which are filled with different road networks in 18c and are followed up with three two-dimensional building structures generated with rules R_2 and R_3, illustrated in the lower row of figure 18.

Figure 19 shows the three-dimensional structures generated on the basis of 18e on the left hand side with rule R_2^{3D}, and on the basis of 18d on the right hand side with rule R_1^{3D}.

4.2. A real-life case study in the "Franzosenviertel"

The three examples illustrated in figures 20-22 are all composed with the same method. At first the size of the cells is defined and the lattice is placed on the site. Next various site developments are created using the field types "Plan", "Cluster", "Vernetzer" and "Ausleger". Afterwards the areas enclosed by roads are filled with the two-dimensional building structures. In the final step, two different three-dimensional building structures are each derived from the two-dimensional structures.

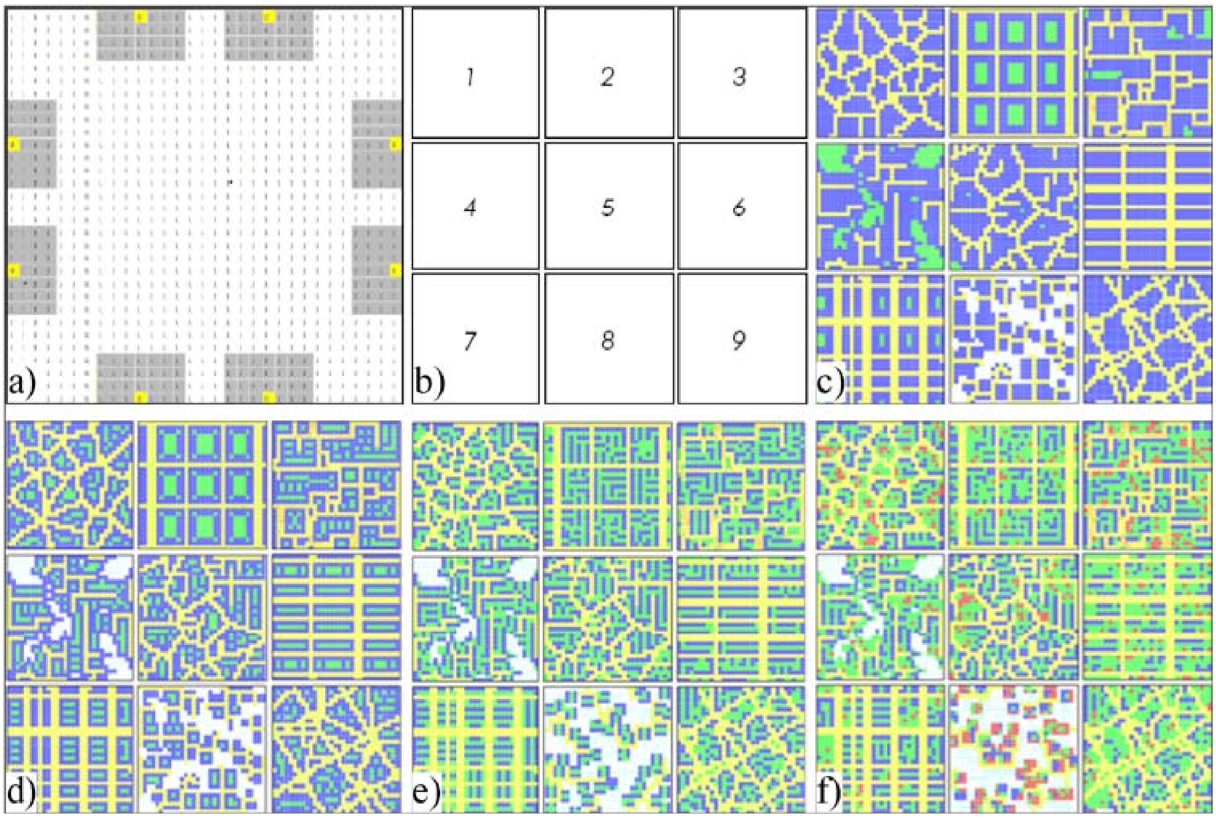

Figure 18: a) Initial configuration; b) Numbering of the fields; c) The fields filled with different road systems: 1. “Cluster”; 2. “Plan” or regular grid; 3. “Vernetzer”; 4. “Ausleger”; 5. “Cluster” combined with “Ausleger”; 6. “Plan” or Rows; 7. “Plan” or irregular Raster; 8. “Ausleger“; 9. “Cluster“ combined with a regular Raster; d) Road system with rule R_2; e) & f) Road system with rule R_3.

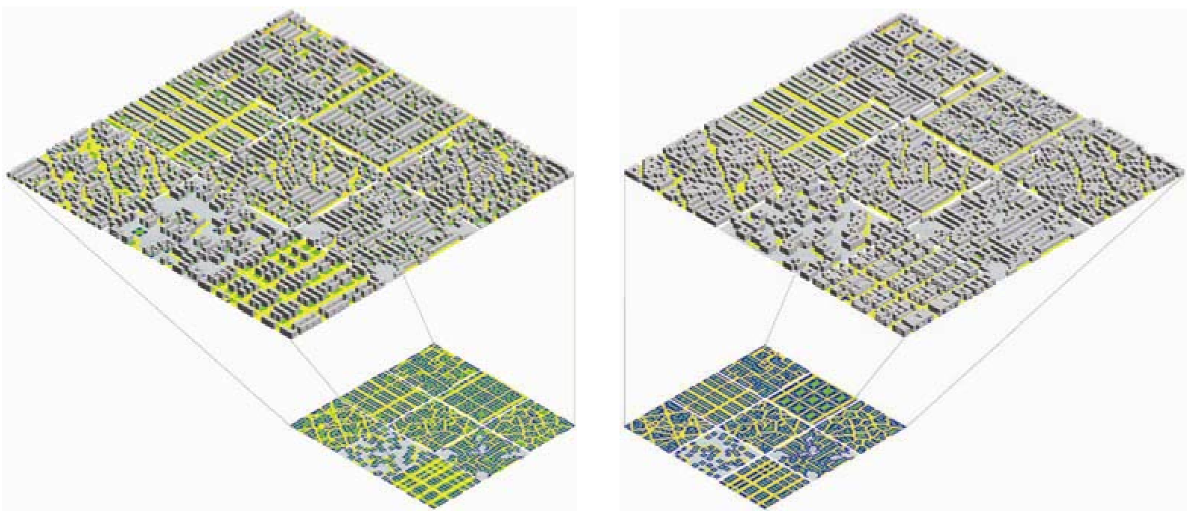

Figure 19: Three-dimensional building structures based on 18e and 18d.

5. Discussion and future prospects

Large construction projects are increasingly developed by private investors only without the participation of public authorities. This is the reason why economic considerations are given priority and social interests such as the quality of urban design are treated secondarily.

In this case we need a strategy that satisfies the investors' interests and nevertheless promotes the architectonic qualities which are of benefit to the general public. For an optimal economic usage of a plot to facilitate profitable buildings the maximisation of the lot coverage and the total floor area has to be ensured. Therefore the computer program developed can be used to meet these conditions and at the same time we can take these requirements as a strategic basis for the design of sophisticated spatial configurations. That is to say that we try to overcome the monotonous standard solutions not only with aesthetic but also with economic arguments. The opinion that architects are made obsolete by the introduction of such technology cannot be upheld as this is a factor of the general mechanisation of the world and its consequences. By comparison we only need consider mathematicians, who have not been eliminated by computer technology but have used it to enhance their science.

Figure 20: Examples for simple rasterised "Plan" structures.

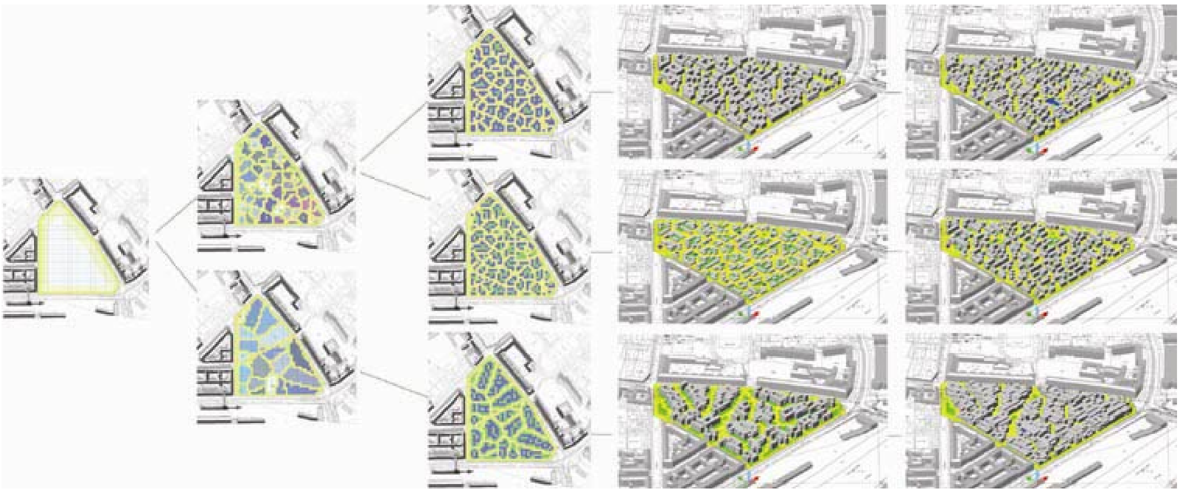

Figure 21: Examples for "Cluster" structures.

Figure 22: Examples for "Vernetzer" (upper row), and "Ausleger" structures.

Economic necessities result in a shortening of the planning period for property development. This generally results in a dependency on standardised solutions and conventional design processes instead of looking for optimal and ambitious concepts. A standardised design process usually makes use of standardised solution patterns instead of searching for an optimal and innovative conception. Quality assurance by a public offer of an idea competition is the exception among private investors.

The support of planning processes using computers as a creative instrument can also improve the rationality as well as the quality of the design. Once the rules, which can also be considered as genotypes, for a structure are explored and proven in practise they can be used for further planning. The result of such a planning process, the phenotype, is different every time, because the generative process depends on the particular local environmental conditions that are expressed by *U(H)* above. In this way it becomes possible to revert to a proven solution without being restricted to a shallow copy. Only the main characteristics of a design are adopted. Artificial strategies of crossing and selecting several successful rules can be combined to achieve further optimised solutions (Goldberg, 1989).

The instruments of the development plan can confirm the land-use and its allocation and provide guidelines for the spatial design by implementing building restriction lines, frontage lines and restrictions of use. At best there was a town-planning competition in the run-up to the development plan for better quality assurance.

A new attempt for establishing a development plan could follow the example of the local counting and voting rules of the cellular automata model. This would offer the possibility to define the phenotype of the buildings by its genotype by means of a specific software system to control the rules and the derivable structures. Spacing distances and building restriction lines could be fixed far more flexibly if one makes the variables dependent on one another and takes into account the initial intention of these regulations to provide each room of a building with sufficient air and light.

Acknowledgement

The Visual Basic files for AutoCAD 2005 and some additional material can be downloaded from: http://www.entwurfsforschung.de/compStadt/compStadt.htm

the NetLogo program is available from:
http://www.entwurfsforschung.de/Strukturfor/netlogo/CA_buildingStructureA.html

References

Alexander C, 1965. "A City is not a Tree" *Architectural Forum* 1 58–62

Batty M, 2005 *Cities and Complexity. Understanding Cities with Cellular Automata: Agent-Based Models, and Fractals* (London, MIT Press)

Batty M, Lonlgey P, 1994 *Fractal Cities: A Geometry of Form and Function* (London, Academy Press)

Coates P, Derix C, Simon C, 2003, "Morphogenetic CA" in *Proceedings of the Generative Art Conference,* Milan, http://www.generativeart.com/papersga2003/a27.htm

Coates P, Healy N, Lamb C, Voon W L, 1996, "The use of Cellular Automata to explore bottom up architectonic rules" in *Eurographics '96* Eds J Rossignac, F Sillion (Blackwell Publishers)

Erickson B, Lloyd-Jones T, 1997, "Experiments with settlements aggregation modesl" *Environment and Planning B: Planning and Design* 24(6) 903–928

Feldt A, 1972 *Community Land Use Game* (New York, Free Press)

Flake W Gary, 1998 *The Computaional Beauty of Nature: Computer Explorations of Fractals, Chaos, Complex Systems, and Adaption* (London, MIT Press)

Goldberg D E, 1989 *Genetic Algorithms in Search, Optimization and Machine Learning* (Addison-Wesley, Boston)

Hillier B, Hanson J, 1984 *The social logic of space,* Reprinted paperback edition 2003 (Cambridge University Press, Cambridge)

Hillier B, Leaman A, Stansall P, Bedford M, 1976. "Space syntax" *Environment and Planning B: Planning and Design* 3(2) 147–185

Humpert K, 1992 *Das Phänomen der Stadt: Berichte aus Forschung und Lehre.* Arbeitsbericht des Städtebaulichen Instituts der Universität Stuttgart, Städtebauliches Institut, Stuttgart

Humpert K, 1997 *Einführung in den Städtebau* (Kohlhammer, Stuttgart)

Krawczyk R J, 2002, "Architectural Interpretation of Cellular Automata" in *Proceedings of the Generative Art Conference,* Milan, 7.1 - 7.8, http://www.generativeart.com/papersGA2002/7.pdf

March L, Steadman P, 1971, *The Geometry of Environment: An Introduction to Spatial Organization in Design* (R.I.B.A. Press, London)

Mitchell W J, 1990 *The Logic of Architecture: Design, Computation, and Cognition,* 6th ed., 1998 (MIT Press, London)

Portugali J, 2000 *Self-Organization and the City* (Springer, Berlin)

Resnick M, 1994 *Turtles, termites, and traffic jams: Explorations in massively parallel microworlds* (Addison-Wesley, Cambridge)

Russel S, Norvig P, 2002 *Artificial Intelligence: A Modern Approach,* 2nd ed. (Prentice-Hall, Englewood Cliffs)

Schaur E, 1992 *Mitteilungen des Instituts für leichte Flächentragwerke: Ungeplante Siedlungen. Non-planned Settlements. IL 39* (Karl Krämer, Stuttgart)

Schweitzer F, 2003 *Brownian Agents and Active Particles: Collective Dynamics in the Natural and Social Sciences:* (Springer, Berlin)

Stiny G, Gips J, 1971, "Shape Grammars and the Generative Specification of Painting and Sculpture" in *Proceedings of IFIP Congress 1971,* 1460–1465

Teichmann K, Wilke J, (Eds) 1996 *Prozess und Form "Natürlicher Konstruktionen"* (Ernst & Sohn, Berlin)

Toffoli T, Margolus N, 1987 *Cellular Automata Machines: A New Environment for Modeling.* (MIT Press, Cambridge)

Watanabe M S, 2003, "The Induction Cities. Induction Design Project", http://www.makoto-architect.com/idc2000/index.htm (16. Januar 2007)

Watanabe M S, 2002 *Induction Design: A Method for Evoutiionary Design* (Birkhäuser, Basel)

Wegener M, 2005, "Urban Land-Use Transportation Models" In *GIS, Spatial Analysis and Modeling* Eds D. Maguire, M. Batty, M F Goodchild, (ESRI Press, Redlands, CA) pp 203–220.

Witten T A, Sander L M, 1981, „Diffusion-Limited Aggregation, a Kinetic Critical Phenomenon" *Physical Review Letters,* 47(19) 1400–1403