SPATIAL AGGLOMERATES

Towards synthetic modelling of the ‘unplanned’

RENEE PUUSEPP
Estonian Academy of Arts, Tallinn, Estonia
renee.puusepp@artun.ee

Abstract. This paper reports on the computational modelling research investigating spatial organisations often associated with unplanned settlements. Such spatial agglomerates are composed of several co-located but autonomous units (agents) that share common facilities and infrastructure (e.g. circulation). Depending on the context, units in the agglomerate represent individual dwellings, apartments or abstract spatial geometry. The paper presents early prototype models that can be interpreted at various scale, and a computational model for generating organic settlement layouts. The originality of the research resides within a new multi-agent algorithm for creating spatial organisations. The agglomeration process benefits from two distinct generative design strategies – self-organisation and adaptive development strategy. While the self-organisation accounts for the emergence of the global structure in the agglomeration, the adaptive development strategy ensures that the basic environmental and spatial requirements of each individual unit are satisfied.

Keywords. Generative design, agent based modelling, object oriented design, unplanned settlements

1. Introduction

Many early settlements display organic spatial organisations of great complexity, yet each building in these agglomerations meet the basic requirements of access, light and space. When looking at a typical European or Middle-Eastern medieval town, we cannot quite make up our minds if these were totally anarchistic and unplanned or quite the opposite – very
carefully planned. This paper suggests a computational method for understanding and recreating the complexity found in such kind of ‘unplanned’ settlements around the world. It is suggested that this method is not only useful for analysing the genesis of ‘unplanned’ settlements in Europe, but also for developing modern housing layouts across the globe. The fitness criteria in the proposed model can be adjusted to reflect the climatological and cultural characteristics depending on the location of deployment. However, the need for sufficient sunlight, space and access are pretty much universal requirements.

The research builds on the hypothesis that the bottom-up process leading to the satisfaction of spatial needs of individual dwelling units can also produce meaningful settlement organisations. This hypothesis is tested by building a computational model using an adaptive development strategy and self-organisation where individual dwellings are modelled as mobile agents. The paper follows a synthetic modelling research by first creating simple prototype models that explore the concept of agent-based space formation. These models are then developed further and the model for generating diagrammatic settlement layouts is proposed subsequently.

Although the synthetic modelling approach to exploring spatial agglomerates by building computational prototypes is relatively unknown to the contemporary architectural practice, the concept has been invented and tested already in 1980’s by Bill Hillier and Paul Coates. Their research shows how the global structure in unplanned settlements emerges solely out of local interactions. Hillier illustrates it with an extremely simple yet conceptually powerful model – the Beady Ring model (Hillier, 1989). The Beady Ring model – or Alpha Syntax as Paul Coates (2010) calls it – was executed with very primitive computational tools that by no means meet the standards or performance of contemporary software tools. Yet, the following generations of designers and architects have largely ignored the findings of the conducted research. The mainstream computational research has been focused on top-down modelling and analysis methods instead. Coates claims that almost all models of urban structure operate at the scale that never deals with the finesse of architectural spaces at the human scale (Coates 2010).

Today, the availability of modern software components makes it possible to develop the original concept to the level where such models can be used for solving practical design tasks. These models can now incorporate advanced parametrical design techniques and environmental analysis methods. The research reported in the proposed paper is a step towards making new computational models available for practicing architects and urban planners.
2. Methodology

Synthetic modelling is seen as a methodology for developing computational models using a simple set of principles towards increasingly complex models that produce sophisticated output. The output of the model is constantly compared to the desired target output and the gap analysis is deployed to inform the direction of further modelling activities. Synthetic modelling facilitates the invention of new methods and techniques that can radically alter the direction of research. Deaton and Winebrake (2000) place synthetic modelling somewhere between synthesis and analysis where the systems are simultaneously built and studied. They suggest using exploratory analysis in order to understand how the system responds to the changed conditions by conducting a series of experiments. The following section outlines several early prototypes and shows how a series of experiments lead to the model for generating settlement layouts.

3. Early prototypes with space forming agencies

Space forming agents do not necessarily need to perform complicated tasks in order to achieve interesting results at the colony level. Simple repulsive behaviour of getting as far as possible from the closest fellow agent in the colony leads to the uniform distribution of agents across the simulated universe and to the emergence of global hexagonal structure (Figure 1).

![Figure 1. Uniform distribution of agents following a simple repulsive behaviour.](image)

With an additional modification, this prototype can be programmed to visualise the space around each agent and to reveal the topological skeleton of uniform yet organic space. In order to achieve this, another type of agents with a stronger repel strength are introduced. Figure 2 illustrates the development process of a new spatial organisation – also known as the Voronoi diagram (Adamatzky 2001). A similar approach of approximating
Voronoi diagrams in collectives of mobile finite automata is described by Adamatzky and Holland (1998).

The simplicity of the repulsion model makes it an ideal starting point to build more intricate prototypes. The abstract nature of the generated skeleton network can be made more tangible when the prototype is redeployed in the context of model-specific constraints. For this purpose agents can be redesigned to recognise and react to additional cues in their environment that are not other agents. For example – agents can be instructed to stay away from certain objects in the model or, on the contrary, be attracted to other elements.

The emergent formation of Voronoi pattern can be computationally heavy depending on the number of agents involved. In order to reduce the amount of computations, a deterministic algorithm (Dey and Zhao, 2003) can be used instead. In contrast to the earlier prototype the next prototype enables the designer to constrain the movement of agents to a custom-shaped bounded region (see Figure 3) and assign the target area that each agent needs to obtain. The agent has two ways of achieving this ‘need’: it can tweak its repulsion strength or change its internal pressure according to the difference between its desirable and actual size. By increasing the repulsion strength the agent pushes other agents further away so that its cell can grow larger; by increasing the internal pressure the agent can push the corners of its cell further away from its nucleus and enlarge its internal area this way. Provided that there is enough available space, an agent can quickly achieve its target size by manipulating these two parameters.
The ability to constrain agents in a predefined region is a useful functionality if one knows the exact shape and the size of the resultant agglomerate. The algorithm works reasonably well if the area matches the accumulated target area of agents, even if it leaves no freedom to the colony to find its own outline shape. However, if one wants to experiment intuitively with different regions or different number of agents, this approach quickly becomes tiresome. The prototype model proposed in the following section features no boundary. The agents are simply placed within a rectangular area and the movement rules and fitness criteria define what shape and size the agglomerate will take.

4. The model for generating ‘unplanned’ settlement layouts

The concept of modelling spatial structures as agencies of individual units driven by their individual ‘needs’ has potentially many implications to the field of generative design. The concept can be used for building agent-based models for architecture and urban design at various scales.

This paper describes the very first experiment that focuses on the task of generating layouts typically associated with unplanned settlements. In many ways, it is too early to make any conclusive claims for the quality of the generated output. The proposed model still relies on many abstractions and the output is optimised to and tested against a limited set of fitness criteria. Therefore, the generated outputs should be treated as informative spatial diagrams rather than thoroughly developed urban layouts. However, the proposed computational model paves the way to a fuller method for generating settlement layouts with intrinsic optimisation of several key characteristics such as accessibility, natural insolation and size of floor area. The model for generating ‘unplanned’ settlement layouts is composed of three distinguishable modules (Figure 4).
Figure 4. The flow diagram of the model featuring three programmable modules and two generative cycles – self-organisation and the larger adaptive development cycle.

The first module is concerned with the self-organisation of individual units and incorporates the movement algorithm that makes the prototype original. The global position of individual dwellings and their relationships with the neighbouring units are dynamically determined during the reiterative process within the module. The second module features a set of parametrical rules for modelling the agglomerate geometry by connecting the neighbouring units together. The third module performs an analysis on the created geometry and assigns a grade value to each unit based on a chosen method of analysis. Whereas all three modules are connected into a repeatable programmatic loop, each module can be also edited separately. Such kind of modular architecture is particularly suitable for the synthetic modelling approach – exploring spatial agglomerates by building computational models.

As mentioned earlier, the first module sets the movement principles of individual agents. It features agent-based self-organisation and can be described the best as an architectural counterpart of the famous Reynolds’ flocking boids algorithm (Macal and North, 2006) that is appropriated to reproduce patterns of common housing layout typologies. Similarly to Reynolds’ algorithm, the global organisation and structure emerges from simple movements of individual agents executed in parallel. The movement rules are explained in Figure 5.
SPATIAL AGGLOMERATES

If the closest neighbour is not within a predefined range ...

RULE 1
Move closer if too far

RULE 2
Move away if too close

... and within the range

RULE 3
Move between the two closest

RULE 4
Move to the opposite side

RULE 5
Move to the perpendicular position

KEY: ● satisfied unit       ○ unsatisfied unit       ⊗ new location

Figure 5. Movement rules of individual agents.

The movement algorithm is capable of generating agglomerations featuring T, H, L, U, Y and other shapes commonly found in housing layouts (Figure 6).

Figure 6. Generated layouts.
The second part of the proposed model takes care of the parametric shape manipulation. Perhaps in reaction to the contemporary obsession with complex geometry and abundance of superficial examples using parametric form finding, this module is the most underdeveloped of the three modules. The shape of the housing layout is simply created by joining neighbouring units together, simplifying the resultant geometry and then recreating individual geometries by subdividing the mass with the Voronoi method. The outcome is then fed forward into the third module.

The analysis module takes the geometry, runs it through a specific quantitative analysis, calculates which units are ‘satisfied’ and can thus maintain their position, and which units should try to find a new and better place. This information is then sent back to the very first module. The values assigned in the analysis module determine how the movement on individuals is adjusted in the next loop – ‘unsatisfied’ individuals become active in finding a new location in the agglomeration.

In order to meet the common prerequisites expected from residential dwellings, the geometry was tested for three different conditions. In the first instance, the individual units were tested for a predefined footprint area. The second test checked whether units receive enough direct sunlight (Figure 7 and 8) and the third one tested for accessibility from the public domain. Only those units that were completely surrounded by other units were considered inaccessible. In each test case, the computational model managed to generate a number of acceptable solutions.

Figure 7. A generated layout showing non-normalised grade values defined by irradiation analysis
The analysis module always operates with the same input format – simple mesh geometry – and produces the same output format – integer values of ‘satisfaction’ assigned to each unit. The consistency of input and output format makes it possible to stack up the tests in order to take multiple criteria into account.

Despite somewhat anecdotal testing criteria and simplistic building geometry, the number of successfully generated solutions proved that the proposed prototype model can indeed produce settlement layouts that satisfy common requirements for individual units such as accessibility, spatial and insolation needs. It is reasonable to argue that the model would also work when the building geometry evolved and more sophisticated analytical methods such as structural and thermal analysis were introduced.

5. Conclusion
All computational models outlined in the paper are examples where the spatial structure was entirely or partly constructed of mobile agents – units that have certain autonomy in finding a geometric configuration to satisfy their own ‘needs’. These space forming agents interact with other agents and together form spatial agglomerates that are easily interpreted as settlement layouts.

The bottom-up approach to creating settlement layouts has two clear advantages over traditional top-down methods. Firstly, the process can be
fully automated and requires no human intervention. Secondly, parallel processes in the agent-based simulation lead to uniformly distributed spatial requirements (e.g. sufficient sunlight) across the settlement layout. This is more difficult to accomplish with top-down methods. However, there is a question whether the bottom-up method would achieve the collective effects such as good street connectivity and protecting public spaces from wind. This can perhaps be done by introducing collaborative behaviour to the agent colony. The workaround for now is to generate multiple solutions and select those that match the desired requirements.

The prototype model for generating ‘unplanned’ settlement layouts shows how computational models can be built in a modular and flexible way. It features three distinct modules, each of which is extendable without the need to modify other modules. For example, the parametric geometry manipulation can be altered by adding new rules for connecting individual dwellings with their neighbours without the need to change neither the analysis module nor the agent-based self-organisation module. Similarly, new analytics can be incorporated or new movement rules introduced without having to worry about breaking the generative cycle.

The research shows how agent-based modelling can be used in generating organically structured settlement layouts. When appropriately combined with other computational methods such as parametric modelling and computational analytics, one can construct complex yet inherently optimised spatial agglomerates. Besides solving common architectural and planning tasks, such kind of models can help devising flexible policies for guiding development processes in modern settlements. One can imagine how appropriate development control mechanisms implemented locally at the level of individual dwellings leads to the consistent and organised development of the entire settlement structure.

References