Self-organised Construction with *Revit*

Sarah Edenhofer, Simon Rädler, Manuel Hoß and Sebastian von Mammen
University of Augsburg
Organic Computing Group
Augsburg, Germany
Email: sarah.edenhofer@informatik.uni-augsburg.de

***Abstract*—Due to innovations in software, robotics and 3D printing, self-organised construction is within reach. It bears great potential for the automatic generation of a wide variety of designs, their integration into the built environment, their structural and automatised optimisation, as well as their dynamic adaptation over long periods of time. In this paper, motivated by the latest empirical findings on the construction methods of social insects, we present a software pipeline for the generation of architectural designs based on self-organisation. A probabilistic, grid-based multi-agent system that implements a flexible stigmergy-based behavioural construction model generates three-dimensional structures. Next, the generated structures could be evaluated in terms of their energy efficiency and these results be fed into an optimisation engine to improve the local behaviours of the construction agents. For visualisation and evaluation of the generated designs we utilise the API of the framework *Revit 2016*[1], a software for Building Information Modelling (BIM) provided by *Autodesk*.**

I. Introduction

Nature is an infinite resource of inspiration for technical system design. An example of nature's ingenuity is the lotus effect, a self-cleaning property of the lotus flower [1], which has been applied to improve industrial surfaces [2]. More abstractly, numerous algorithms and (meta-)heuristics used by software developers are inspired by nature, for example genetic algorithms, bee algorithms, particle swarm optimisation, simulated annealing, ant colony optimisation, harmony search, or firefly algorithms [3], [4]. Self-organised construction can be observed in the nest construction efforts by social insects [5]. One key property of these processes is the lack of a central entity that concerts all its peers. Another aspect is the entities' local coordination by means of stigmergic cues in the environment. Mason [6] describes how the placement of pheromones triggers the construction of different nest shapes. There is even research conducted correlating ants' and termites' building behaviours with the evaporation rates and intensities of the underlying stigmeric pheromones cues [7]. To some extent, approaches of self-organised construction have already been applied in the domain of robotics. An example for an according ground robot is the marXbot [8], [9], a light-weight robot with the ability to sense obstacles and holes in the ground and to interact with building blocks by means of two magnetic grabs. Airborne robots, such as retrofitted quadrocopters, have the advantage of also being able to move vertically [10]. Both ground and airborne units for self-organising construction are still mainly prototypes, struggling with issues such as load and energy limitations, accuracy in movement and deployment of construction material. In the next section, we describe the overall approach to self-organised construction design and its current state of implementation. We conclude with an elaboration on the next steps that we need to take.

[1] www.autodesk.de/products/revit-family/overview

II. A Software Pipeline for Self-organised Construction Design

The basic idea of our approach is to let virtual agents coordinate their construction efforts. Similarly to [11], [12], the interplay of local behaviours would result in the development of three-dimensional morphologies. These emerging constructions would then be evaluated with respect to various quantifiable criteria, which are, in turn, used to optimise the underlying construction behaviours. As we are primarily concerned with generating architectural designs, we would like to consider criteria such as energy efficiency, structural integrity, usable space, incidence of light, similarity to or continuity in the built-environment, etc. This pipeline of (1) agent-based self-organised generation of morphologies, (2) their computational evaluation, and (3) the subsequent optimisation of local construction behaviours requires various design decisions such as the kind of the utilised optimisation technique. Due to the wide range of parameters and especially the open, generative character of the task (large numbers of solutions are not only possible but desirable), we resort to Evolutionary Algorithms, and more specifically to Genetic Programming [13], for specifying and optimising the virtual construction agents' behaviours.

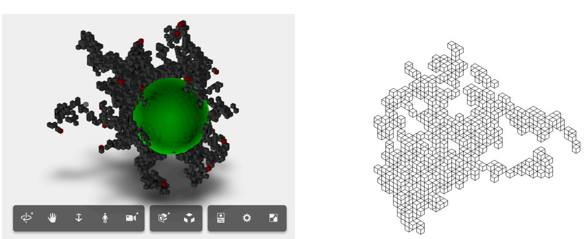
A. Model Cornerstones

Our virtual model agents roam in space freely (also vertically) in a 3D lattice grid. They can place or remove construction material in their immediate Moore-neighbourhood, i.e. in the 26 cells around them. Only one building block or one agent can occupy any single cell at any point in time. Pheromones, however, are deposited independently and blended accordingly. Their diffusion is modelled by means of arbitrary functions dependent on proximity and time. Should our model, at some point, require the consideration of more complex obstacles hindering signal spread or gain from GPU-based shader processing, we might implement an alternative model relying on grid-based diffusion-reaction [14]. The agents act asynchronously, i.e. the execution loop iterates all the agents consecutively but in random order. The agents' behaviour is encapsulated in one method which may make use of an agent's own state as well as of descriptive data about its environment, e.g. locations, kinds, and intensity of pheromones and occupied and vacant grid cells close-by.

Similarly to the pheromone secreted by the queen of a termite hill [5], an initial *queen pheromone* is placed in the simulation at the beginning to serve as a global point of reference. At each step, the agents analyse the pheromones in their environment. This analysis determines their behaviour, whether to add or remove one or more components and where to build or move, next. Certain signals may, e.g., trigger the agent to lay down a building block in its vicinity. Additionally, the agent might deposit a pheromone on top of the building component and move one step further. Currently, we have queen pheromones and (de-)construction pheromones implemented, this set of pheromones is expandable arbitrarily.

B. Pipeline Challenges So Far

We started to implement our simulation model with the cloud-based *Fusion360*[2] software because of its flexibility and its promising feature set (Figure 1(a)). However, our first prototype revealed that at the time of this writing, it is not easily possible yet to setup the pipeline we need in order to evaluate the generated building designs in terms of energy efficiency. More specifically, we could not pass the building information to an according evaluation engine at this point. Therefore, we switched to the Autodesk Revit platform. As a first step, implementing a *Revit Direct Shape Project*, our agents create test morphologies solely performing random walks (Figure 1)(b)). Next, the Revit Project would be sent to the Cloud, with the goal of performing a Voxel-Based Energy Analysis by means of Green Building Studio[3]. The result of the analysis would then be used to calculate the fitness value of the considered building design. Afterwards, new agent behaviours and thus new building designs would be generated using a Genetic Programming (GP) algorithm. Over time, GP would explore the tandem of behaviour-genotypes and building-phenotypes and yield a set of different optimised solutions.



(a) (b)

Fig. 1. Self-organising construction built and rendered using (a) Python/Fusion360 and (b) C# and Revit. The queen pheromone is visualised as a green sphere, the agents in red, wandering about around and deposited building blocks in grey.

[2]www.autodesk.de/products/fusion-360/overview

[3]www.autodesk.com/products/green-building-studio/overview

III. Next Steps

In order to avoid long evaluation times, we need to complete the implementation of automated model analysis interfacing with the Revit cloud. Before that, we need to make sure that our model meets the minimum requirements for analysis, including material assignments and building specifications. In case there are incompatibilities between our grid-based model and the energy efficiency solver, we might introduce intermediary processing steps to create few smooth surfaces. We expect the GP to evolve new behaviours of self-organising construction agents to run locally based on the retrieved simulation results. A fast browser for identifying interesting architectural designs would be mandatory to use the bred designs as starting point for human-built architecture.

Acknowledgment

The authors would like to thank Philipp Müller and Jeremy Tammik for their help and support.

References

[1] A. Lafuma and D. Quéré, "Superhydrophobic States," *Nature Materials*, vol. 2, pp. 457–460, 2003.

[2] A. Solga, Z. Cerman, B. F. Striffler, M. Spaeth, and W. Barthlott, "The Dream of Staying Clean: Lotus and Biomimetic Surfaces," *Bioinspiration & Biomimetics*, vol. 2, no. 4, pp. 126–134, Oct. 2007.

[3] X.-S. Yang, *Nature-Inspired Metaheuristic Algorithms*. Luniver Press, 2008.

[4] M. Dorigo and T. Stützle, *Ant Colony Optimization*. Bradford Company, 2004.

[5] S. Camazine, N. R. Franks, J. Sneyd, E. Bonabeau, J.-L. Deneubourg, and G. Theraula, *Self-Organization in Biological Systems*. Princeton, NJ, USA: Princeton University Press, 2001.

[6] Z. Mason, "Programming with Stigmergy: Using Swarms for Construction," in *Proceedings of the Eighth International Conference on Artificial Life (ICAL)*. MIT Press, 2003, pp. 371–374.

[7] A. Khuong, J. Gautrais, A. Perna, C. Sbaï, M. Combe, P. Kuntz, C. Jost, and G. Theraulaz, "Stigmergic Construction and Topochemical Information Shape Ant Nest Architecture," *Proceedings of the National Academy of Sciences*, vol. 113, no. 5, pp. 1303–1308, 2016.

[8] S. Magnenat, R. Philippsen, and F. Mondada, "Autonomous Construction Using Scarce Resources in Unknown Environments," *Autonomous Robots*, vol. 33, no. 4, pp. 467–485, 2012.

[9] T. Soleymani, V. Trianni, M. Bonani, F. Mondada, and M. Dorigo, *Autonomous Construction with Compliant Building Material*. Springer International Publishing, 2016, pp. 1371–1388.

[10] S. von Mammen, S. Tomforde, J. Hähner, P. Lehner, L. Förschner, A. Hiemer, M. Nicola, and P. Blickling, "Ocbotics: An organic computing approach to collaborative robotic swarms," in *Swarm Intelligence (SIS), 2014 IEEE Symposium on*. IEEE, 2014, pp. 1–8.

[11] S. von Mammen, C. Jacob, and G. Kókai, "Evolving swarms that build 3d structures," in *CEC 2005, IEEE Congress on Evolutionary Computation*. Edinburgh, UK: IEEE Press, 2005, pp. 1434–1441.

[12] S. von Mammen and C. Jacob, "Evolutionary swarm design of architectural idea models," in *Genetic and Evolutionary Computation Conference (GECCO) 2008*. Atlanta, USA: ACM Press, 2008, pp. 143–150.

[13] R. Poli, W. B. Langdon, N. F. McPhee, and J. R. Koza, *A field guide to genetic programming*. Lulu. com, 2008.

[14] A. R. Sanderson, M. D. Meyer, R. M. Kirby, and C. R. Johnson, "A framework for exploring numerical solutions of advection–reaction–diffusion equations using a gpu-based approach," *Computing and Visualization in Science*, vol. 12, no. 4, pp. 155–170, 2009.