Emergent networks: A slime mold simulation

Niklas B. Semmler
6191177

Bachelor thesis
Credits: 9 EC

Bachelor Opleiding Kunstmatige Intelligentie

University of Amsterdam
Faculty of Science
Science Park 904
1098 XH Amsterdam

Supervisor
Prof. dr. P. Sloot

Informatics Institute
Faculty of Science
University of Amsterdam
Science Park 904
1098 XH Amsterdam

June 24th, 2011

Abstract

In a world where news are delivered in an instant around the world, where the population of a network service is around twice the one of the USA and where an natural catastrophe can lead to a chain reaction a reliable infrastructure is the key to survival. This insight has motivated a diverse group of researchers to seek understanding of the resilient infrastructure of nature surrounding our everyday life.

One life-form has especially attracted attention in this context: The slime mold. This little unicellular organism is able to create, without any central control, transport networks with extraordinary properties. Biological experiments by Tero et al [28] have shown, that the slime mold *Physarum polycephalum* could create a network complex enough to compete with the railway system in the area of Tokyo. Such and similar results have spawned a number of projects aiming at the simulation of the slime mold's behavior [27, 9, 26, 12]. Yet at present no universal explanation has emerged. The present paper tackles this problem. The slime molds behavior is analyzed and general rules are extracted. Further a model effort in the Cellular Potts Model [8] and the CompuCell3D [6] framework is described. Consequently the simulation is compared to biological experiments. The highlight of the paper will be a presentation of the slime mold simulation on a map of the Netherlands.

1 Introduction

The process of modeling biology with computational means is nearly as old as the computer itself. Even Turing, one of the frounding fathers of computing, invested his last years to recreate patterns occurring everywhere in nature, the so-called Turing patterns [32], with the help of simple reaction-diffusion mechanisms.[1] On this line the attention the slime mold received is just a logical consequence. It is a unicellular organism without any central organ for information processing, yet it is able to interact intelligently with its environment. Consequently every life stage of the slime mold's life cycle has attracted researchers. Be it its movement [15] or its culmination [14]. Recently the plasmodium stage, most prominently expressed by the *Physarum polycephalum*, has come under the spotlight. In this stage the slime mold explores its environment and creates a network between all located food sources [22, 24, 20, 21, 19]. This plasmodium network is indispensable for the slime mold's survival.

[1] That Turing was keen to simulate the simplest of biological behavior should not be a surprise to us: "Instead of trying to produce a programme to simulate the adult mind, why not rather try to produce one which simulates the child's?" [31]

Just as its human counterparts the mold is able to create transport systems of astonishing size and complexity. Yet it requires only local decentralized forms of interaction lacking any kind of central nervous system [28]. Simple experiments have shown that it approximates optimal graphs like the Steiner Minimal Tree [20]. In this light researchers have wondered whether the slime mold could be used as a tool to design smarter transport systems [5, 3, 28, 2]. Such a work with the slime mold has however severe limitations. Not only does it take hours to find a stable path, but the constraints which can be put on the environment are limited. To solve all of these problems, a simulation seems feasible. Especially so as other stage's of the life cycle have been successfully simulated [15, 14].

First attempts have every reason to make us feel optimistic. Gunji et al proved that some of the slime mold behavior could be recreated by a simple cellular automaton [9]. Tero et al approximated the slime mold's way to solve a maze [27]. Jones et al went a step further including movement, growth, shrinkage and oscillation in his particle model implementation [12, 29, 11, 30].

That these efforts from seemingly unrelated sides have independently (partially) succeeded seems astonishing and promising. Yet at the same time we might wonder how they will affect future research. All these models have only small groups of support in a very small research community. To make the most of it a common framework and understanding seems advantageous. One of the prospects for a commonly used framework is the Cellular-Potts-Model [13, 8]. One of its most advanced implementation, the CompuCell3D environment [10]. Not only do they include a wide variety of mechanisms far beyond the scope of a typical cellular automaton (PDE's, volume constraints, etc.), but other parts of the slime mold's life cycle have already been modeled with success [14].

In the present paper I will present first results about the possibilities of the framework to model mentioned behavior. In section 2 I will reduce it to a small set of rules which guided the implementation. In the following section 3 both the Cellular Potts Model and the CompuCell3D framework will be highlighted. Further in section 4 design decisions will be explained. Section 5 compares the simulation to the biological substrate and reports on an effort to obtain a tube network for a map of the Netherlands. Next, section 6 reviews the success and shortcomings of the project.

2 The slime mold

In the plasmodium life stage, the *Physarum polycephalum* is an amoeba-like organism with a broad tubular network. Thanks to its unicellular nature it is commonly perceived as a rather simple organism [19]. To describe even the behavior of the

"simple" slime mold in quantitative terms is, however, increasingly complex. Instead I will follow the path Herbert A. Simon, another pioneer of computation and AI, has pointed out a decade ago: "As we seek to develop theory in computer science, and specifically in AI, we should be looking for laws of qualitative structure and regularities of organization and process that characterize them." [25]

From an eagle eye perspective we can find the slime mold to be a computation trying to create a viable network for communication and transportation under a set of constraints of its environment. These constraints appear as either attracting (food, favorable temperature, etc) or repulsing (toxics, enemies, illumination, etc.) fields around the slime mold. Consequently a network emerges between the foodsources which we can approximately picture as a planar graph [1].

The main mechanism of the slime mold is its two part tube structure. On the one hand there are the spontaneously assembling/disassembling actin-myosin fibers, i.e. the tubes, creating an oscillatory pressure. On the other there is the passive plasmodium which transports chemicals and food particles in waves through the tubes [14]. These two parts intertwine closely. The flow of the plasmodium steers the direction of the fibers, while the direction of the fibers limits the flow of the plasmodium.

Most of its mass lies at the food sources (hitherto referred to as FS) to maximize the food intake. Food particles are extracted periodically and the FSs serve as oscillators triggering bidirectional streaming (also referred to as shuttle streaming). The remaining part of the organism develops dependent on the environmental conditions. If conditions are favorable it moves omni-directionally with a huge number of fine tubes between the outer membrane and the FSs. If conditions are less pleasant it advances with thick tubes in the most favorable directions [26]. The influence of the environment seems to be primarily on the membrane (weakening or strengthening it) and secondarily on the strength of contraction.

For the development of the tubes a number of rules have been discovered:

1. longer, thinner, open-ended tubes disappear [21]
2. tube size is bound (or saturating) [27]
3. flow of plasmodium through a tube increases it [27]
4. food particles are distributed by chance [24]
5. amount of food effects only time scale of development [23]

3 The framework

As stated in the introduction the framework employed here was the Cellular Potts Model (hitherto CPM). The CPM is based on the Ising model, which was utilized to explain magnetic fields in ferromagnetic materials. Since 1920 it has received a number of transformations so that it shares only little resemblance with the original model. (For a more throughout outline of its history see [8].) Over the years it has been implemented multiple times for different purposes. Yet only recently have general frameworks been devised. Here the CompuCell3D has been chosen as it appeared as the most advanced and supported simulation.

3.1 Cellular Potts Model

The CPM is build around the idea that a complex biological situation can be approximated by an energy minimization function, a Hamiltonian ($\mathcal{H}$). The Hamiltonian relates every possible configuration of a grid to a single energy value. Cells or cell-like objects (represented as σ) are entities occupying a number of sites on the grid. The function restricts interaction of these entities by a variable number of constraints. One of the basic constraints is adhesion and cohesion:

$$\mathcal{H} = \sum_{(\vec{i},\vec{j})neighbours} J_{\tau,\tau'}(\sigma_i,\sigma_j)(1-\delta_{\sigma_{\vec{i}},\sigma_{\vec{j}}}) + \sum_{(\vec{i},\vec{j})neighbours} \frac{J_{\tau,\tau'}(\sigma_i,\sigma_j)(\delta_{\sigma_{\vec{i}},\sigma_{\vec{j}}})}{2}$$

$J_{\tau,\tau'}$ represents the contact energy between two types of cells (τ and τ'). The delta term ($\delta_{\sigma_{\vec{i}},\sigma_{\vec{j}}}$) is a Kronecker delta being one in case that both sites are of the same cell (used to separate inter-cell adhesion from intra-cell behavior). The coordinates on the grid are given by the vectors $\vec{i}$ and $\vec{j}$ respectively. Hence the algorithm will calculate the current energy by identifying what types of cells are connected in what manner.

There are many additional ways to define constraints on the Hamiltonian. Volume, perimeter/surface or length constraints can limit the cell shape by using constraints of the form:

$$\mathcal{H}' = \mathcal{H} + \lambda_\tau(v_\sigma - V_\tau)$$

Here V_τ stand for the ideal volume of the cell type and v_σ for the actual volume of one specific cell. λ_τ specifies the importance of the constraint.

Another class of constraints can be used to restrain the directions in which a cell expands:

$$\mathcal{H}'' = \mathcal{H}' - \mu F(\vec{x})$$

F is the directional field and μ defines the importance of the field and therefore the strength of the reaction to this constraint. In this way elongation or chemotaxis can be employed.

Several other constraints are possible such as connectivity constraints [18, 16] or even grid independent constraints like cell velocity and viscous flow [4].

When the constraints have been chosen, the CPM will follow a variant of the Metropolis algorithm to minimize the energy. It will try to change the cell of one random site (σ_{target}) to one of its neighbors (σ_{source}). Whether this process succeed depends on the amount of the energy difference between the current and the new configuration: $\Delta\mathcal{H} = \mathcal{H}_{final} - \mathcal{H}_{initial}$. If the difference is smaller than an offset or yield Y it is accepted. If it is bigger its acceptance depends on a Boltzmann probability and the fluctuation amplitude T:

$$p\left(\left(\sigma\left(\vec{i}_{target}\right) = \sigma_{target}\right) \rightarrow \left(\sigma\left(\vec{i}_{target}\right) = \sigma_{source}\right)\right)$$
$$= \begin{cases} 1 & if \Delta\mathcal{H} < -Y \\ e^{\frac{-(\Delta\mathcal{H}+Y)}{T}} & if \Delta\mathcal{H} \geq -Y \end{cases}$$

A number of such "copy attempts" (usually equal to the grid size) counts as one Monte Carlo Step (MCS). Steps are repeated until a threshold (in the simplest case a time limit) is met.

3.2 CompuCell3D

The CompuCell3D [6] software was developed to ease the writing of CPM simulations. Most of the simulations can be implemented with a simple XML or Python script. If more customization or higher speed is necessary, it is possible to define individual addons in C++.

There are two classes of addons. Steppers are employed after every transition in the CPM. Steppables on the contrary are used after every MCS. Steppers serve the implementation of new constraints. Steppables are more often employed to define environmental conditions as for example external fields.

CompuCell3D is primarily used for its cellular automaton but it is very easy and straightforward to implement a partial differential equation to include e.g. the behavior of chemicals (see the following subsection).

Its richness of expression has allowed several different simulations. One famous and simple demonstration, the cellsort algorithm, is illustrated in the appendix (see figure 1).

3.3 Chemical Fields

As expressed above, the slime mold can be seen as a computation on a field of attractants and repellents. To model such field the compucell environment entertains the possibility of including partial differentiation equations. The form of these fields is three-fold: Secretion, Diffusion and Decay.

$$\frac{\partial c}{\partial t} = \alpha \delta_{\sigma_x,0} + D \bigtriangledown^2 c - (1 - \delta_{\sigma_x,0}) \epsilon c$$

Here α, ϵ are constants of the respective process. D is the diffusion matrix. The Kronecker delta term $\delta_{x,y}$ is used to limit the secretion or decay only to some cells but not others (the term becomes one only if $x \neq y$). In this special case the secretion will appear only in the cell and decay only in the Medium. While this equation (inspired from Gamba and Serinin in [17]) might not exactly represent the technique employed by the CompuCell3D environment it gives the reader the tools to follow the further reasoning.

4 Model

After the biological behavior and the framework for implementation had been fixed, a bridge had to be built to match one with the other. This necessarily entails the abstraction and sometimes neglect of parts of the biology. Essentially the algorithm was split into three distinct elements. One, the environment posing the computational question to the organism in form of food sources secreting attractants and obstacles emitting repellents. Two, an autocatalytic loop responsible for the network structure of the slime mold. Three, the growth and oscillatory behavior of the cell's volume creating a fitness pressure.

4.1 Direction: Attraction and Repulsion

That the slime mold would react to any chemical indicating food (called ATTR) or unfavorable conditions (referred to as REP) appears self-evident from an evolution-

ary perspective. The question is only how the chemicals might influence the slime mold. In literature (see [7]) use of the LALI principle (local activation long-range inhibition) has led to considerable results. Yet in this case the primary objective of the slime mold must be to arrive at all potential food sources. The evasion of obstacles on the path can only be of secondary importance. Hence I have preferred LILA (local inhibition long-term activation). Consequently food sources have been modeled with high diffusion rates and relatively low decay rates of ATTR where repellents (foremost used as walls) have been modeled with low diffusion rates and relatively high decay rates of REP. Additionally the secretion and chemotactic reaction is higher for ATTR and lower for REP to reflect the relative importance of food.

4.2 Structure: Autocatalytic loop

When one is trying to create a network structure in such a distributed system one faces two difficulties. Either the cell disperses into small autonomous sub-cells or it moves like an indistinguishable blob. The first risk is circumvented by disallowing disconnections of the cell (see [18, 16] for details). The latter has to be met with the implementation of an autocatalytic loop. This part is a substantial element of the existing simulations (mesmerized flow in [9], learning rule in [27] or chemical release in [12]). It corresponds to the tube-plasmodium interaction in the real organism discussed earlier (see section 2). In the current model it is treated by a higher Medium adhesion than cell cohesion and, similar to the model of Jones [12], by making the cell release additional attractants. The lower adhesion to the Medium pushes the cell to explore its environment. In the following this first differentiated parts of the cell create a field of gravitation (by releasing ATTR) pulling more and more mass in their direction.

4.3 Optimization: Growth and Oscillation

Left to itself the mechanisms described above will produce a network spanning over all of the food sources (given that they are not to distant). However the resulting graph of the network will be non-optimal including a great amount of redundancy. When to much of the mold amasses at any single spot its release of ATTR will create its own center of gravity competing with the food sources. As described in the last section blob formations are penalized, yet this can be bypassed by forming an arbitrary number of small loops. To arrive at more realistic results a certain fitness function must be employed. In the real slime mold there are two indications of such a function. At the one side the oscillation of the actin-myosin fibers or tubes and at the other the growth and shrinkage of the molds volume.

Oscillation is meant to close the smaller loops in the process. The fluctuation should overstrain the small loops so that they are unable to stabilize and create long lasting inefficiencies. In the same way this behavior can weaken the expansion of the mold distorting the envelopment of more remote food sources.

The growth/shrinkage behavior counteracts this phenomenon. It pressures the mold to evolve from a dynamic exploration to a determinate crystallization. In the current version this is done by regulating a part of the ideal volume (see section 4) in a linear fashion. Additionally the slime mold's volume is increased for every connected food source.

4.4 Minor changes

Other than the principles outlined above a number of smaller changes have been implemented. To mirror the non-infinitely small size of molecules more closely, the effect of the chemicals was configured as to saturate at a certain level. Also a random number of food sources increase their ATTR secretion every 50 MCS for the duration of one MCS. This way they appear less homogeneous and hence more realistic.

Still the effect of these changes is perceived, though not extensively tested, to contribute only slightly.

5 Results

In the following results of the simulation will be presented. These can be separated into two categories. At first the simulation will be compared with results from biological experiments recorded in literature. Then the simulation will be analyzed on a greater level.

5.1 Biological plausibility

Two experiments are deemed especially noteworthy in the light of current simulation effort. Results of experimentation by Takamatsu have shown that the slime mold responds actively to its environment adapting its structure to maximize chances of survival. If the repellents predominate the cell will form stronger, fewer and more distinct tubes to protect its interest. In the opposite condition, the attractants overweight the repellents, the mold will evolve smaller pathways and its structure will appear rounder, approximating a circle. (For an excellent depiction please review [26].) To inspect whether the simulated mold would behave in the same way it was exposed to a field with four foodsources and, optional four sources of repulsion. The results (see figure 2) point in the right direction. Here too, does

the absence of repulsory forces initiate a broader exploration of the environment. Their existence in contrast restricts the mold to just a few strong pipelines.

The other experiment which has to be referred to is one of the many conducted by Nakagaki. He claimed that the mold's behavior can be approximated as a bistable system of the Steiners Minimal Tree and a cyclic graph [20]. To put this to a test the simulated mold was brought onto a field with three food sources arranged in a cycle. In one condition the mold was initially placed in the middle of the dish, in the other it was placed next to a food source on the border. In the first he formed what can clearly be recognized as a variant of the Steiner Minimal Tree. In the second, when starting from the side, he included two loops. (See figure 3 in the appendix.) While these are not versions of cyclic trees, it can be seen as a general tendency going down the same path.

A great number of further experiments could be simulated to try this algorithm. An example would be the famous maze solved by the mold [23]. For the moment these two have to suffice.

5.2 Macro-Behavior

The potential of the simulation would seem rather restricted if it was not to scale up to more than as few as three or four foodsources. Even more so as the real mold has already been employed to recreate road and railway maps of Tokyo, the UK, the USA and the Netherlands. To give the simulated mold a similar challenge a new abstraction of the dutch railway system including 21 cities was devised. Restrictions like the border to the sea or other nations have been modeled by a repellent secreting wall (see 4). The slime mold had to explore the area while growing and manifest a stable network while shrinking. Greater distances between some foodsources and some odd arrangement of walls as for example the lower right area have added to the difficulty. Yet the mold was able to build a complete graph at most of the time. (Only in a few simulations the upper left foodsource was excluded for some strange reason.)

One of the last solutions (figure 5) has been transformed to a graph (figure 8). (In this case the connection to the upper left foodsource has been artificially added, see figure 6.) Two characteristics of the graph have been computed for comparison with literature. The total length of the graph normalized by a Minimum Spanning Tree (figure 7) is: $TL_{MST} = 1.17$. The fault tolerance (FT) was calculated by estimating the probability that a part of the graph becomes disconnected at the occurrence of one (FT_1) or two (FT_2) break-ups at random edges: $FT_1 = .6257, FT_2 = .3747$. The edges where in all cases weighted by their respective length.[2] The cost-benefit ratio ($\alpha_N = FT_N/TL_{MST}$) was correspond-

[2]A simple example: In a triangle a single broken line will not lead to any distortion. Hence

ingly: $\alpha_1 = .5347, \alpha_2 = .3202$.

6 Discussion

It would go to far to believe an accurate replication of the biologic behavior was accomplished. However this should not surprise us. While I would expect the simulation and the principles above to be a useful approximation it is an abstraction nevertheless. As stated above a number of phenomena have been neglected. Other enormously simplified.

For example does the simulation clearly lack more complex interactions such as shuttle streaming or the spread of food particles. In fact the contact with the foodsources seems at all implausible. The real organism would employ most of its mass to envelop the foodsources and it would never at all cost move away from them risking rather to be split. Similarly the algorithm is far away from the elegance of the biology. In the virtual environment the path between the foodsources is fluctuating and changing directions. At some points it seems rather noisy (which might be attributed to the the probabilistic nature of the model). The real thing in contrast changes its pathways only slightly after initial growth. The differentiation happens more by competition between existing tubes than a reorganization of single ones.

If anything the digital mold can be compared to the biological substrate in a similar manner as most artificial neural networks to their biological partner. They are nowhere near their real counterpart and still show similarities in behavior. They provide the necessary insights to push our understanding of biology even further. Finally they may find their place in the toolbox of the engineer instead of remaining a mere reflection of nature.

In one aspect the simulation might however be very similar to the real. At least if we were to assume that the current modeling effort on the map of the Netherlands is comparable to the one encountered by Tero et al. In this case we would find that the simulation has a similar cost-benefit ratio as the real slime mold or a railway network [28]. Different to this research a comparison with the railway system has been avoided here. The reason is simple. The details which are included in a single train map are hard to grasp and even harder to implement. We cannot, at current state, handle things so complex as geography and the particular

it's fault tolerance is 1. ($FT_1 = 1$) If one line is omitted ever any further distortion will lead to a disconnection. Therefore the fault tolerance is 0. ($FT_1 = 0$) In a complexer graph where the breaking of some but not all lines will lead to a disconnection, the probability that a line is broken which leads to no consequences is summed. When the fault tolerance against multiple accidents is measured, the probability of lines which can safely break together are multiplied and all of the results are summed up. For a more detailed analysis see [20]

flow of people between cities (work, holiday, etc). The utility is the understanding and inspiration gained to create new kinds of networks and to create them from a different perspective.

With all criticism it should not be forgotten that a particular strength of the simulation is the employed framework. Thanks to the task-independent language of the CompuCell3D framework it is a simple matter to change or expand the simulation to meet the problems expressed above. It should also be noted that even the current simulation offers numerous ways of optimization. The parameters which were estimated in this case could be explored with a machine learning algorithm in far greater depth.

The doors opened by the current work outweigh its draw-backs by far.

7 Conclusion

The present work was performed in four steps. First the behavior of the mold was examined and, so far as possible, transformed into qualitative rules. Then the means in form of the Cellular Potts Model and its implementation the CompuCell3D framework were described. Both parts restricted subsequent modeling efforts. The complex behavior of the slime mold was reduced to three general principles. Experiments with this model have led to considerable results. Biological observations were partly met and the simulation was able to create a graph over all cities of the netherlands. Characteristics of this graph were found to be similar to earlier results [28].

The current model is the second to include movement and oscillation with the path finding capabilities. (For the other by Jones et al see [12].) It is the first to embrace a general framework.

8 Acknowledgements

I would like to thank Peter Sloot for his guidance and trust on this project. Also I want to thank Jeff Jones for his quick response and information over the details of his algorithm.

References

[1] A. Adamatzky. *Physarum machines*. World Scientific, 2010.

[2] A. Adamatzky and R. Alonso-Sanz. Rebuilding iberian motorways with slime mould. *Biosystems*, 2011.

[3] A. Adamatzky and J. Jones. Road planning with slime mould: If physarum built motorways it would route m6/m74 through newcastle. *Arxiv preprint arXiv:0912.3967*, 2009.

[4] A. Balter, R.M.H. Merks, N.J. Popławski, M. Swat, and J.A. Glazier. The glazier-graner-hogeweg model: extensions, future directions, and opportunities for further study. *Single-Cell-Based Models in Biology and Medicine*, pages 151–167, 2007.

[5] D.P. Bebber, J. Hynes, P.R. Darrah, L. Boddy, and M.D. Fricker. Biological solutions to transport network design. *Proceedings of the Royal Society B: Biological Sciences*, 274(1623):2307–2315, 2007.

[6] Indiana University Biocomplexity Institute. Compucell3d. `http://www.compucell3d.com`, 2011. [Online; accessed 13-June-2011].

[7] E. Bonabeau. From classical models of morphogenesis to agent-based models of pattern formation. *Artificial Life*, 3(3):191–211, 1997.

[8] J.A. Glazier, A. Balter, and N.J. Popławski. Magnetization to morphogenesis: a brief history of the glazier-graner-hogeweg model. *Single-Cell-Based Models in Biology and Medicine*, pages 79–106, 2007.

[9] Y.P. Gunji, T. Shirakawa, T. Niizato, and T. Haruna. Minimal model of a cell connecting amoebic motion and adaptive transport networks. *Journal of theoretical biology*, 253(4):659–667, 2008.

[10] J.A. Izaguirre, R. Chaturvedi, C. Huang, T. Cickovski, J. Coffland, G. Thomas, G. Forgacs, M. Alber, G. Hentschel, S.A. Newman, and J.A. Glazier. Compucell, a multi-model framework for simulation of morphogenesis. *Bioinformatics*, 20(7):1129, 2004.

[11] J. Jones. Characteristics of pattern formation and evolution in approximations of physarum transport networks. *Artificial Life*, 16(2):127–153, 2010.

[12] J. Jones. Influences on the formation and evolution of physarum polycephalum inspired emergent transport networks. *Natural Computing*, pages 1–25, 2010.

[13] A.F.M. Marée, V.A. Grieneisen, and P. Hogeweg. The cellular potts model and biophysical properties of cells, tissues and morphogenesis. *Single-Cell-Based Models in Biology and Medicine*, pages 107–136, 2007.

[14] A.F.M. Marée and P. Hogeweg. Modelling dictyostelium discoideum morphogenesis: the culmination. *Bulletin of mathematical biology*, 64(2):327–353, 2002.

[15] A.F.M. Marée, A.V. Panfilov, and P. Hogeweg. Migration and thermotaxis of dictyostelium discoideum slugs, a model study. *Journal of theoretical biology*, 199(3):297–309, 1999.

[16] R.M.H. Merks, S.V. Brodsky, M.S. Goligorksy, S.A. Newman, and J.A. Glazier. Cell elongation is key to in silico replication of in vitro vasculogenesis and subsequent remodeling. *Developmental biology*, 289(1):44–54, 2006.

[17] R.M.H. Merks and J.A. Glazier. A cell-centered approach to developmental biology. *Physica A: Statistical Mechanics and its Applications*, 352(1):113–130, 2005.

[18] R.M.H. Merks, S.A. Newman, and J.A. Glazier. Cell-oriented modeling of in vitro capillary development. *Cellular Automata*, pages 425–434, 2004.

[19] T. Nakagaki, M. Iima, T. Ueda, Y. Nishiura, T. Saigusa, A. Tero, R. Kobayashi, and K. Showalter. Minimum-risk path finding by an adaptive amoebal network. *Physical review letters*, 99(6):68104, 2007.

[20] T. Nakagaki, R. Kobayashi, Y. Nishiura, and T. Ueda. Obtaining multiple separate food sources: behavioural intelligence in the physarum plasmodium. *Proceedings of the Royal Society of London. Series B: Biological Sciences*, 271(1554):2305, 2004.

[21] T. Nakagaki, H. Yamada, and M. Hara. Smart network solutions in an amoeboid organism. *Biophysical chemistry*, 107(1):1–5, 2004.

[22] T. Nakagaki, H. Yamada, and M. Ito. Reaction-diffusion-advection model for pattern formation of rhythmic contraction in a giant amoeboid cell of the physarum plasmodium. *Journal of theoretical biology*, 197(4):497–506, 1999.

[23] T. Nakagaki, H. Yamada, and A. Toth. Path finding by tube morphogenesis in an amoeboid organism. *Biophysical Chemistry*, 92(1-2):47–52, 2001.

[24] T. Nakagaki, H. Yamada, and T. Ueda. Interaction between cell shape and contraction pattern in the physarum plasmodium. *Biophysical Chemistry*, 84(3):195–204, 2000.

[25] H.A. Simon. Artificial intelligence: an empirical science. *Artificial Intelligence*, 77(1):95–127, 1995.

[26] A. Takamatsu, E. Takaba, and G. Takizawa. Environment-dependent morphology in plasmodium of true slime mold physarum polycephalum and a network growth model. *Journal of theoretical biology*, 256(1):29–44, 2009.

[27] A. Tero, R. Kobayashi, and T. Nakagaki. A mathematical model for adaptive transport network in path finding by true slime mold. *Journal of theoretical biology*, 244(4):553–564, 2007.

[28] A. Tero, S. Takagi, T. Saigusa, K. Ito, D.P. Bebber, M.D. Fricker, K. Yumiki, R. Kobayashi, and T. Nakagaki. Rules for biologically inspired adaptive network design. *Science*, 327(5964):439, 2010.

[29] S. Tsuda and J. Jones. The emergence of complex oscillatory behaviour in physarum polycephalum and its particle approximation. *Artificial Life*, 12:698–705, 2010.

[30] S. Tsuda and J. Jones. The emergence of synchronization behavior in physarum polycephalum and its particle approximation. *Biosystems*, 103:331–341, 2010.

[31] A.M. Turing. Computing machinery and intelligence. *Mind*, 59(236):433–460, 1950.

[32] A.M. Turing. The chemical basis of morphogenesis. *Philosophical Transactions of the Royal Society of London. Series B, Biological Sciences*, 237(641):37, 1952.

A Pictures

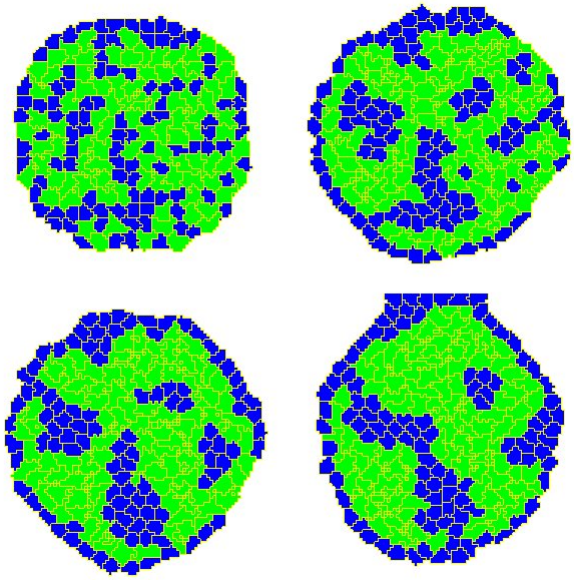

Figure 1: Cellsorting by low-level interaction, a demonstration from the CompuCell3D environment.

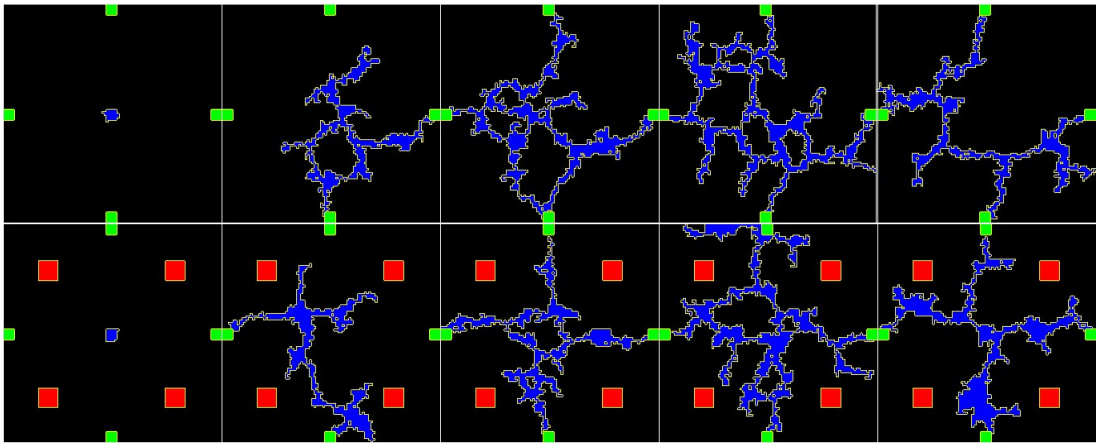

Figure 2: The slime mold on a map with four foodsources with and without sources of repulsion. The five pictures correspond to 1,2500,5000,7500 and 10000 MCS respectively. The grid size is 100×100 pixel.

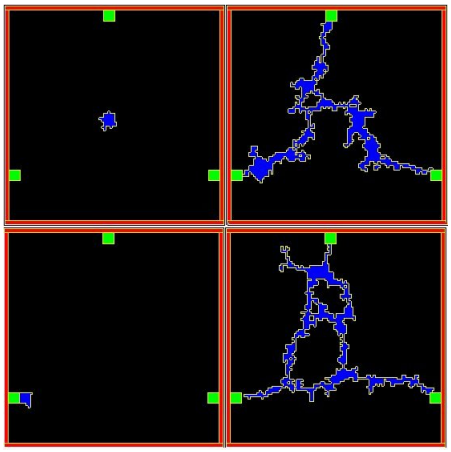

Figure 3: The mold with three foodsources arranged in a cycle starting either in the middle or at the side at 1 and 10000 MCS respectively. The grid size is 100×100 pixel.

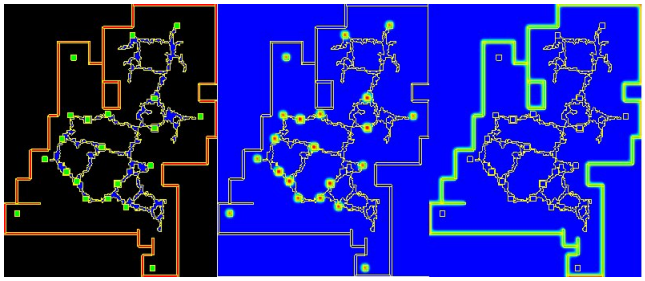

Figure 4: The three layers of a simulation: the cellular automaton and the attractive and repulsive field respectively. The grid size is 200×300 pixel for the x and y dimension respectively.

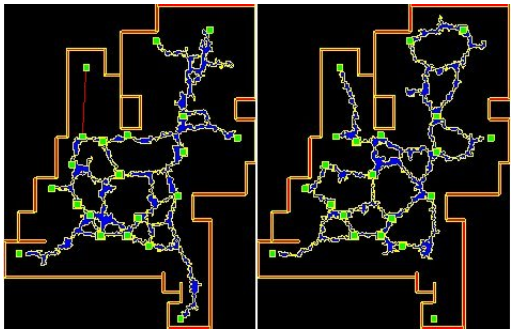

Figure 5: The latest configuration of the mold with an path added and an earlier configuration.

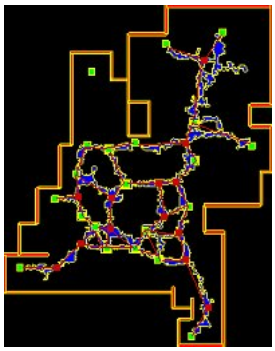

Figure 6: An graph transformation of the mold done with GIMP.

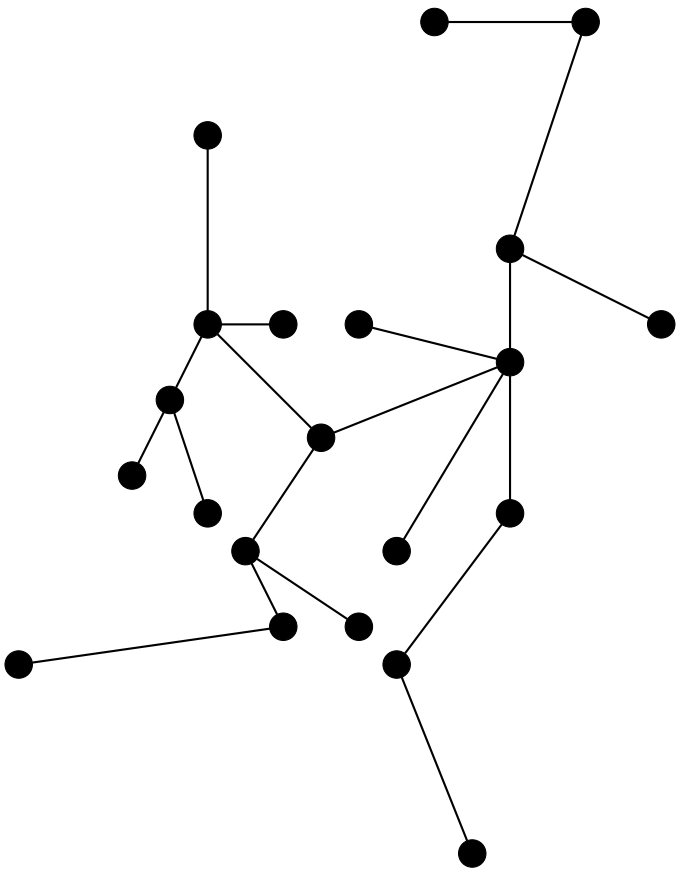

Figure 7: The Minimum Spanning Tree

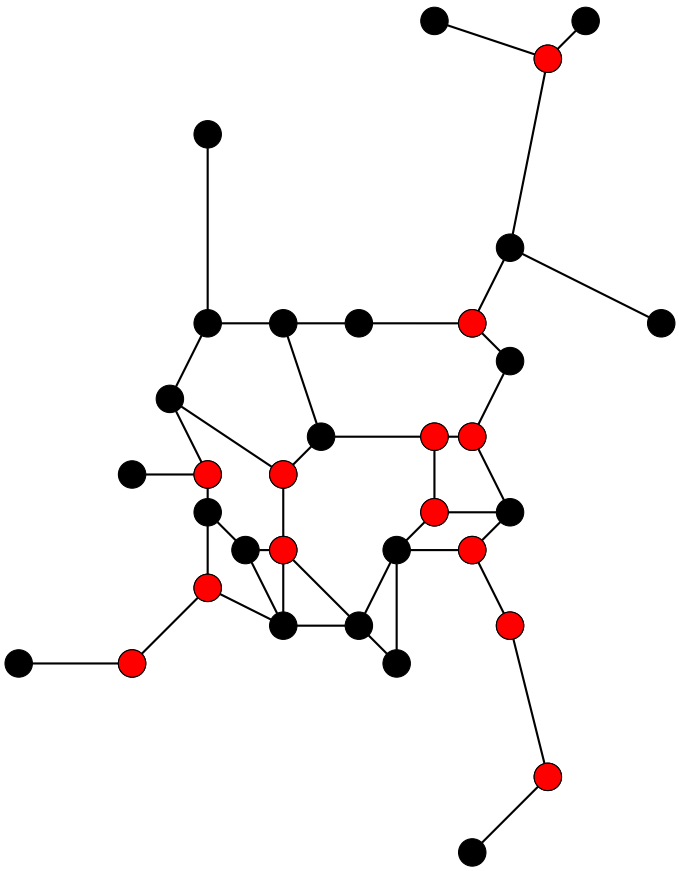

Figure 8: The graph extracted from the mold (red nodes are non-foodsource nodes added by the slime mold)