

Modelling Self-Organising Networks With Slime Mould *Physarum Polycephalum*

Agnes Cameron

Abstract

The slime mould *Physarum Polycephalum* has been shown to exhibit graph-solving [5, 20, 19], computational [2, 9] and complex decision-making [1, 5, 6, 10, 15, 19] behaviours, whilst consisting only of a simple single-cellular organism. Through analysis of cellular automata simulations of *Physarum*, we can investigate the emergent behaviours required to generate self-organising swarm networks. Many current models of *Physarum* behaviour do not lend themselves to distributed and self-organising network analyses [7, 19, 20]. Here, the ‘Vacant-Particle’ model for *Physarum* migration [5] is used as the basis for a true cellular automaton, implementing critical network-forming behaviours such as chemotaxis [19] in a decentralised model. Network formation is analysed across scales in both randomly-initialised and regularised forms, and a basic implementation of distributed self-optimising behaviour is proposed. A comparison is made between the simulation and the behaviour of a living *Physarum* colony. Potential applications to self-organising networks are discussed.

1 Introduction

1.1 Executable Biology

There is a growing trend in biological modelling toward ‘executable biology’ [3], using computational models of biological systems to break the natural world into modular interactions, which may then be analysed as a set of discrete states. In particular, the study of emergence – the phenomenon by which complex group behaviours ‘emerge’ from the interaction of simple individual agents – may be used to model systems of otherwise uncomputable complexity [3, 5, 22].

In [3], a tension is defined between the conventions of so-called ‘dynamic graph’ models that analyse systems globally from the top down, and ‘computational/algorithmic’ models, which reduce the system to an instruction set, and seek to determine its behaviours from the bottom up. The interplay between micro- and macroscopic views of biological systems is also discussed by Gunji et.

al. [5], who propose that the true ‘answer’ to determine the behaviour of a biological system lies somewhere between these two conceptions.

1.1.1 Cellular automata

Cellular automata are an early demonstration of the emergent properties of decentralised systems, using a static grid of states inhabited by mobile agents to map the effects of local rules on global pattern formation. Perhaps most famously, Conway’s Game of Life – which models a changing population of ‘living’ and ‘dead’ cells using a set of only 3 governing rules [4] – spawns a set of complex behaviours.

In [22], Wolfram poses cellular automata as the means by which neg-entropic systems (those which become less chaotic over time) may be analysed and observed. This ‘self-organisation’ makes cellular automata attractive in modelling nonlinear swarming [16], oscillatory [10] and decentralised [16, 22] systems. Swarm systems are particularly attractive from both a situated computing and an executable biology viewpoint as they display complex behaviour with simple agents that are grounded in a local viewpoint.

1.2 Morphological computation in *Physarum Polycephalum*

The amoeboid true slime mould *Physarum Polycephalum* is a unicellular organism comprised of an outer membrane, a plasmodium fluid and a distribution of nuclei, from which the organism draws its ‘many-headed’ (polycephalic) title [9, 10]. *Physarum* is of great interest to mathematicians, computer scientists (and, occasionally, philosophers), due to its ‘intelligent’ properties; namely forms of decentralised [10], parallel [2, 9] and ‘brainless’ computation [9, 15].

A compelling [8] demonstration of *Physarum*’s graph-solving properties was presented by Tero et. al, who showed that, given an array of food sources in the layout of the Tokyo subway stations, the plasmodium formed a near-perfect copy of the real subway network [20], appearing to ‘calculate’ the minimum distance between each node.

Jones [9, 10], provides an overview of the applications of *Physarum* in the field of ‘unconventional computing’, a term coined to include fuzzy, soft and nonlinear computing architectures [9]. In [17], asynchronous logic gates are modelled using a living *Physarum*, demonstrating Boolean algebra using a living organism.

1.3 Models for *Physarum* Behaviour

Models of *Physarum* behaviour tend to fall into the categories defined by Fisher and Henzinger [3], with an initial interest from the mathematical community driving a trend in ‘dynamic graph’ analyses of minimum-distance networks [10, 12, 19, 20, 7]. Gunji et. al [5] propose the first cellular

automaton model of *Physarum* to include the ‘sol-gel’ transformation¹ critical in amoebic motion: since then, several cellular automata have been proposed that seek to model different elements of *Physarum* behaviour [6, 11, 18, 21].

Tero et. al propose an explanation for this behaviour in the form of a ‘dynamic graph’ model for *Physarum* optimisation over food sources [19, 20]. In this model, the plasmodium is abstracted to an array of tiny tubes, which are initialised as evenly distributed across a finite area containing a number of food sources. This is a model of a reaction-diffusion network. Each food source is modelled as a node within the network, oscillating between ‘source’ and ‘sink’ states. The tubes are composed of an Actin-Myosin fibril wall (cytoskeleton ‘gel’) encasing a cytoplasmic ‘sol’. The optimisation mechanism proposed is a positive feedback loop between the flux of sol between nodes, and conductivity of each tube. As more food material flows between nodes, the Actin-Myosin fibres in these tubes are stiffened and expanded, causing the tube radius to increase. Thicker tubes have a higher volumetric flow rate, and thus will continue to grow thicker, where other tubes will die off.

A common theme amongst *Physarum* models of network optimisation is to ‘pre-initialise’ the plasmodium as filling an area, and then observe tubes connecting nodes thicken, whilst unused tubes die away [6, 10, 11, 19, 20]. This forms a realistic model of the ‘final form’, but does not model critical foraging behaviours. By contrast, Gunji et. al, and Tsompanas et. al [5, 21] outline generative approaches to model *Physarum*’s changing morphological distribution both during network formation, and as ‘food signals’ are applied to different parts of an established colony. Both models are shown to solve complex graphical problems, including Spanning Trees, Moroni Diagrams and mazes [5, 21]. Gunji et. al [5] connect the sol-gel transformation observed in *Physarum* colonies with the morphology of *Physarum* networks, using a cellular automaton to demonstrate observed properties of the system.

1.3.1 *Physarum*-inspired networks

Physarum-inspired networks have been used as a basis for fault-tolerant [7], scalable [21], and self-organising [18, 21] designs.

Algorithmic approaches to fault tolerance inspired by *Physarum* are explored in Houbraeken et. al [7], who use the nodal analysis techniques proposed in [19] as a model for dynamic network bandwidth allocation modulated by the rate of information flow. The algorithm proposed in [19] is extended by iterative selection of ‘source’ and ‘sink’ nodes, between which directional flow is defined. A clear parallel is drawn between the rate and volume of flow and the idea of ‘bandwidth’ of a networked system, and the efficiency of the network formed measured against the flow rate between established nodes.

Both Song et. al and Tsompanas et. al [18, 21] use *Physarum*-inspired agent-based models as the basis for self-organising designs for wireless sensor networks (WSNs). WSNs are networks of

¹the transition between fluid cytoplasm (sol) and stiff cytoskeleton (gel)

autonomous nodes that are used to detect information about an environment. Agent-based systems are of particular interest in the design of WSNs, as analytical optimisation of WSN architectures is computationally expensive [21].

This project seeks to model network formation in *Physarum* using a true cellular automaton based on the VP-transportation model described in [5], and explore the potential applications of these models to scenarios similar to those described in [18, 21].

2 Constructing CA models of Physarum

Here we explore the potential for expanding an existing agent-based model of *Physarum* proposed by Gunji et. al [5] to investigate *Physarum* network formation. A model of chemotactic behaviour is applied to the basic 'Vacant-Particle' (VP) model [5], and rudimentary self-optimising behaviours based on graph models in [7, 19, 20] are introduced. The properties of the model are tested according to exploration-exploitation trade-offs, and the effectiveness of the self-optimising behaviour is analysed in comparison to existing models [19, 21]. This leads to a discussion of the use of *Physarum* simulations as a model for self-organising networks across scales, where 'scale' is here defined as the number of nodes that exist for a network of different size.

2.1 Initial Model

Gunji et. al define VP-transportation as a mechanism for network formation in [5], incorporating the 'sol-gel' transformation as a cellular automaton. In this model, *Physarum* 'cells' may occupy one of two states: either as an external 'cytoskeletal' cell, or internal 'cytoplasmic' cell [5]. The plasmodium cells act as mobile agents within a 50x100 simulation grid. The original simulation consists of two phases, a 'growth phase' and a 'migration phase'.

Here, the grid cells are defined as $C_{i,j}$, which may be in state S_0 (empty), state S_1 (cytoplasm sol), or state S_2 (cytoskeleton gel). All changes to a cell $C_{i,j}$ take place according to the state of cells in the neighbourhood, $C_{i,j(nb)}$, of that cell. The neighbourhood is defined according to a Von Neumann neighbourhood, and consists of $C_{i,j+1}$, $C_{i+1,j}$, $C_{i,j-1}$ and $C_{i-1,j}$.

In the stable state, S_1 cells may never border S_0 cells, as the plasmodium is contained at all times within a cytoskeletal S_2 wall. Initially, time $t = 0$, with the state of each cell at time t given by $C_{i,j}^t$.

2.1.1 Growth

In a rudimentary initialisation stage, the plasmodium expands radially outwards, forming an agglomeration approximate to the first stages of spreading from the sclerotium² [5]. With each cycle, the cytoskeleton expands outwards such that, for each $C_{i,j}^t$:

²The clumped dormant state of *Physarum*, which, when hydrated, will activate and grow outward

$$C_{i,j}^{t+1} = \begin{cases} 2 & \text{if } C_{i,j}^t = 0 \text{ and if any of } C_{i,j(nb)}^t = 2 \\ 1 & \text{if } C_{i,j}^t = 2 \\ 0 & \text{otherwise} \end{cases}$$

2.1.2 Vacant-Particle transportation

In the migration phase, empty cells are absorbed by the softening of the cytoskeleton ('sol-gel' transformation) and transported through the cytoplasm via VP-transportation [5].

At the start of each cycle, an cell in state S_2 swaps places with a randomly-chosen neighbouring empty cell S_0 . This 'vacant particle' $v_{i,j}$ is subsumed into the plasmodium as a bubble, its previous position now occupied by a cytoskeleton cell. At this point, the entire cytoskeleton is modelled as flowing cytoplasm: a softening mechanism proposed to aid the process of foraging [5] termed the sol-gel transformation.

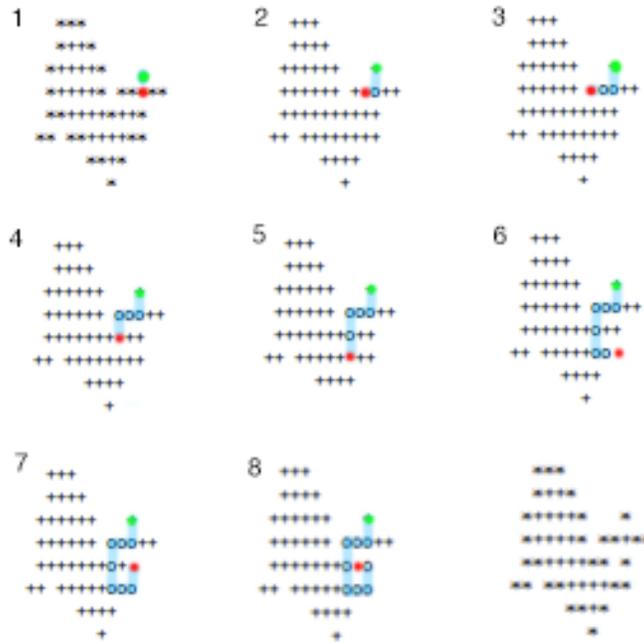
$v_{i,j}$ migrates through the plasmodium cell-by-cell, swapping places with adjacent S_1 cells, leaving a trail of chemorepellant (putting the cells in temporary state S_3) behind it, that stops it from revisiting cells. This motion through the plasmodium will continue until either m cycles have elapsed (where m is a constant defining path length), or the cycle is terminated as $v_{i,j}$ becomes trapped or reaches the edge of the plasmodium. $v_{i,j}$ is defined as having reached an 'edge' when three of $C_{i,j(nb)} = S_0$. $v_{i,j}$ is trapped if all of $C_{i,j(nb)}$ are either S_3 or S_0 .

2.2 Chemotaxis

The model proposed in [5] takes a purely morphological approach to migration, where sections of the plasmodium are crudely selected to become entry 'sites' for VPs. Whilst this produces realistic network formations in restricted settings (e.g. maze solving), in order to produce useful models of network formation in free space, a decentralised model is proposed. In order for this model to be a true cellular automaton, *Physarum* cell interaction is defined in terms of the local environment. *Physarum* has been shown to respond to a chemoattractant gradient under experimental conditions, due to the presence of sensor proteins within the membrane which trigger the stimulation of metabolic processes [21]. The VP entry is redefined as taking place over a probabilistic distribution determined by the chemoattractant concentration in each cell ($CHA_{i,j}$), which is a function of the distance $dist_x$ from food source f_x .

Here, the chemoattractant gradient has a $1/r^2$ distribution across neighbouring cells, an equivocation to network signal strength discussed in 5.1. The signal is constructive – thus chemoattractant signals from food sources will add, increasing the probability of a VP-event in cells close to multiple sources.

Figure 1: Schematic representation of VP-Transportation



In step 1, a VP $v_{i,j}$ (red) changes place with a S_2 gel cell (green). In step 2, after the entry of $v_{i,j}$, the entire plasmodium is rendered as S_1 state cells (+ symbol). Steps 2-7 show the migration of $v_{i,j}$ through the plasmodium, each step swapping places with a neighbouring S_1 cell, leaving a memorised flow-path of S_3 temporary states (blue). This migration is terminated when $v_{i,j}$ becomes trapped within its own flow (8), causing a void to appear. After the sequence is terminated, the plasmodium re-stabilises, with all S_1 states now bordering an S_0 cell rendered as S_2 (* symbol). Thus, the plasmodium has 'migrated' one cell in the direction of VP-entry. This process repeats every cycle.

$$\begin{aligned}
dist_x &= ||f_x - c_{i,j}|| \\
CHA_{i,j} &= \sum_{x=1}^F \frac{k}{dist^2}
\end{aligned}
\tag{1}$$

Where k is a normalising constant across the grid size. The probability of a randomly-selected cell $C_{i,j}$ becoming the site for a VP-entry $V_{i,j}$ is directly proportional to $CHA_{i,j}$, and thus a net migration in the direction of food sources, F , is observed. A graphic of this model is shown in fig. 2.

2.3 Active nodes

A food source $F_{i,j}$ is defined as an ‘active node’ $N_{i,j}$ if one or more of $F_{i,j(nb)}$ is in state S_1 . Once a source is active, the plasmodium does not withdraw, and is continuously surrounded by at least one gel-state cell [19]. This mechanism ensures that the node will retain at least 1 S_1 state element each cycle, adding them if they disappear (analogous to growth with food sources).

3 Scalability

Here, the chemotactic behaviour of the plasmodium is analysed for both randomly and non-randomly initialised simulations of the model proposed in 2.2. The simulations test the number of cycles required to cover all the available food sources in a particular initialisation, testing the scalability of network formation.

3.0.1 Non-Random Initialisation

The non-random configuration consists of a set of nodes distributed in a radial configuration around a central node, with all nodes but the central one falling outside the central quadrant. The sclerotium is initialised upon the central node, distributed evenly in every direction.

3.0.2 Random Initialisation

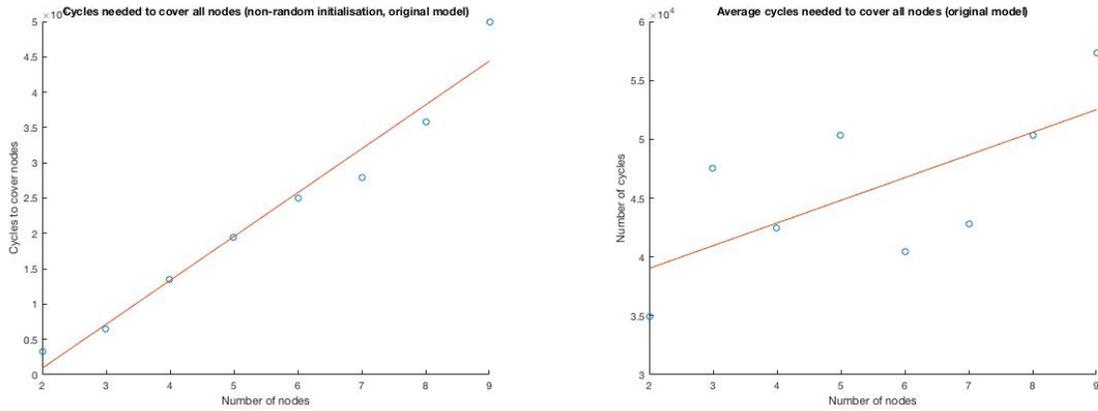
The random configuration initialises the sclerotium at any cell on the grid, as a randomly-initialised blob of ‘seed cells’. The nodes are distributed randomly throughout the grid.

Figure 2: Overview of *Physarum* chemotaxis



The migration and the formation of a network over a randomised initial set of 5 randomised food sources (purple), after 10,000, 20,000, 30,000 and 40,000 steps, for an initial seed size of 60, with 8 growth steps

Figure 3: Scalability across initialisations



(a) Non-random initialisation

(b) Random initialisation

The average number of cycles to 'find' all nodes, for the initial model of the plasmodium, with an initial seed of 60, and an 8-step growth phase

3.1 Variation with number of nodes

In order to test the scaling properties of the regularised and randomly-initialised *Physarum* configurations, the average number of cycles to cover all available nodes in a finite area is measured in fig. 3. Both figures show an approximately linear relationship between the number of available nodes, and the number of cycles taken to spread to those nodes. The randomised initialisation (fig. 3b) requires more cycles on average to discover all nodes for each initialisation than the regularised one (3a). However, the increase in the average number of cycles to find all nodes as the number of nodes increases is less for the randomly-initialised model.

4 Cellular self-optimisation

A critical element of *Physarum* network behaviour is self-optimisation [7, 19, 20], which establishes a positive feedback loop between inter-nodal flow, and the path length between these nodes. Here, a mechanism for approximating the feedback *Physarum* described in [19] using a cellular automaton is proposed and analysed according to a measure of the 'path length' between nodes over time.

4.0.1 Self-optimising algorithm

If two or more active nodes $N_{i,j}$ are present in the network, then for each node in turn, a 'flow particle' $p_{i,j}$ is initialised. $p_{i,j}$ progresses through the plasmodium according to the same 'memorised flow' rules established in 2.1.2, recording its path in a vector $P_{i,j}$. The flow is terminated if the particle comes to an edge, or becomes trapped (as in 2.1.2), or if $p_{i,j}(nb)$ contains another node $N'_{i,j}$

(a 'successful' termination). In the latter case, a cytoskeletal cell in the flow path recorded by $P_{i,j}$ becomes the site for a VP-entry (as in 2.1.2), and the normal mechanism for VP transportation is implemented, causing a unit of growth to occur in the flow-path between the two nodes. The length of the path, $P_{i,j}(\text{length})$ between the nodes is recorded. If the termination is unsuccessful, the process is repeated until a successful termination occurs (up to 500 iterations).

4.1 Measuring self-optimisation

Self-optimising behaviour is here analysed in both the original and the updated model. The measurement of the degree of self-optimisation is here defined as the ease by which plasmodium may flow between network nodes, and is measured by the average path length $P_{av}(\text{length})$ travelled by a flow particle $p_{i,j}$ between established nodes $N_{i,j}$, for every stage in the cycle. This analysis was performed for the non-random initialisation, and an average of the path length between all active nodes each cycle, for systems initialised with 2-9 nodes, are recorded in fig. 4.

Fig. 4 shows an initial sharp decrease in average node-to-node path length over time for both models, over all nodes. Each model is tested over configurations with 2-9 available sources. The path lengths for 2-5 nodes in the initial model (fig. 4a) are stable after 14,000 cycles, with path lengths for the 6- and 7- node models steadily decreasing, and those for the 8- and 9- node models steadily increasing. For the updated model (fig. 4b), the average path lengths are stable for the 2- and 3-node models, steadily decreasing for 4-, 5-, 6-, and 7-node models, and increasing in the 8- and 9-node models. The net path length at 14,000 cycles is less across all nodes for the self-optimised model (an average of 5.4% shorter), and the net number of models where the path length continues to decrease is greater for the self-optimised model.

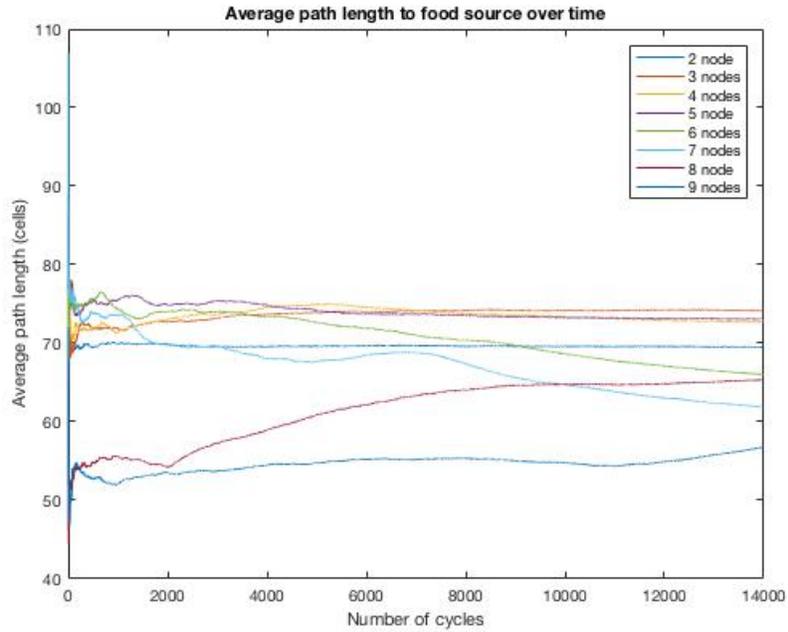
4.2 Scalability Comparison

The experiment conducted in 3.1 is repeated in order to compare the scalability of the initial and updated models across different numbers of nodes. The average time to cover a set of randomly-initialised nodes is plotted against the number of nodes (fig. 5). The gradient of the curve plotted in fig. 5b is significantly steeper than that of fig. 5a, showing a greater increase in time to span a full network as the number of nodes increases in the updated model. The average amount of time taken to find all the nodes is also greater for the updated model.

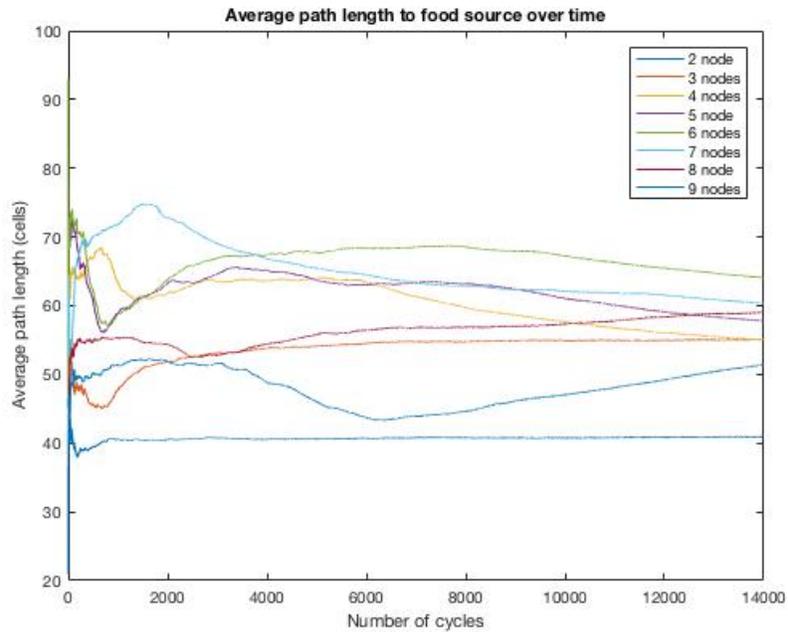
5 Comparison with Real Physarum

In order to compare the model of *Physarum* growth with that of a real sample, a living *Physarum* sample was initialised in the same state as the 5-node, 'regularised grid' model (fig. 7), and photographed at 12-hourly intervals (6-hourly for the first day) over a period of 5 days (fig. 6). The

Figure 4: Variation in inter-nodal path length over time



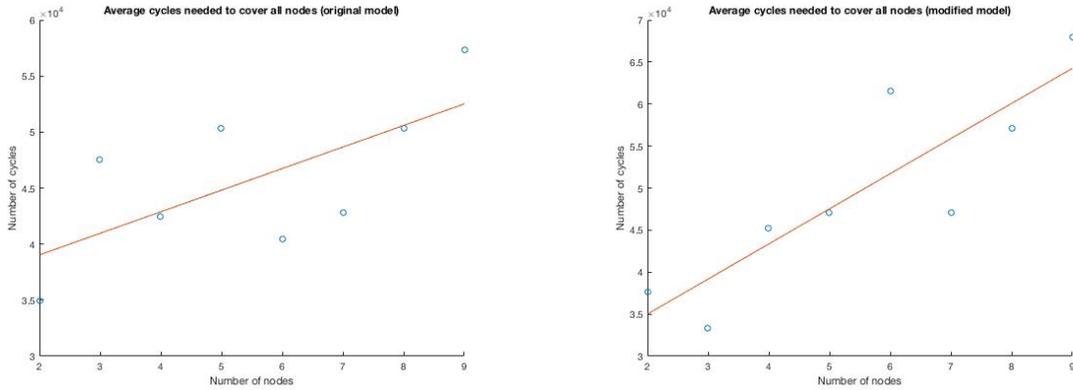
(a) *initial model*



(b) *updated model*

The average path length $P_{i,j}(\text{length})$ required for a flow particle $p_{i,j}$ to migrate between nodes, for simulations with 2-9 nodes, non-random initialisation, seed size 70, growth steps 4

Figure 5: Variation in number of cycles to find all nodes with number of randomly initialised nodes



(a) Initial Model

(b) Updated model

The average number of cycles to ‘find’ all nodes, for a plasmodium with an initial seed of 60, 8-step growth phase

model’ was re-created using a grid of 4 oats on an agar substrate³ spaced 4cm apart, with a fifth oat in the centre upon which the plasmodium was initialised. Every 12 hours, the sample was also given 2ml of water to prevent dehydration, and was cultured at room temperature, in a darkened cupboard.

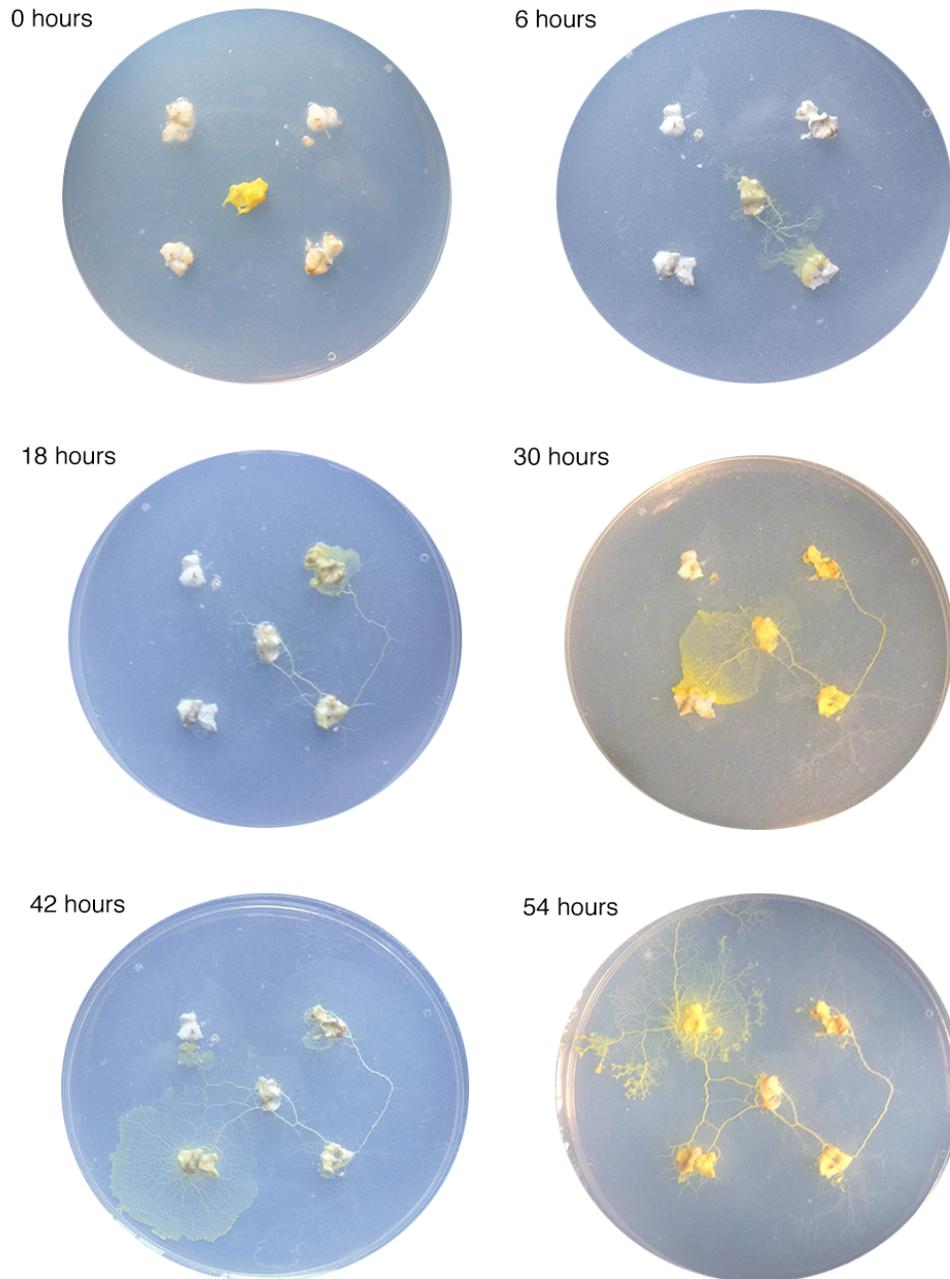
5.1 Observations

Both the real and simulated model are characterised by a near-linear variation in the time taken between establishing each new node, within every 12 hours in the real sample, and 6000 cycles in the simulation. The morphologies of the two networks, however, are distinctly different, with the real *Physarum* forming fine ‘tubes’ between sources, whilst the simulation establishes a more ‘blob-like’ configuration. This is partly a product of the granularity of the simulation: with a grid of only 5000 cells to cover, the variation in thickness appears far more extreme. As can be seen between 42 and 54 hours in the real *Physarum* sample, the tubes between nodes thicken and become more defined, demonstrating the ‘bandwidth increase’ described in [7, 19, 20].

The other main difference established between the samples is that of foraging behaviour, with the real sample demonstrating a dendritic searching pattern (characterised in [1, 10]) that branches out into unexplored areas until a strong chemotactic signal can be established. This is not demonstrated by the simulation, which instead iterates based only on chemoattractant distribution.

³The preferred food source for *Physarum*, sterilised to prevent bacterial growth

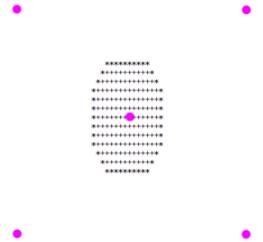
Figure 6: Real *Physarum* colony growth



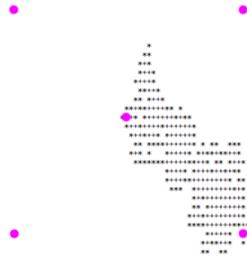
Sample of Physarum Polycephalum grown on 5% agar at room temperature, using sterilised oats as a food source, using the regularised node pattern used in the 5-node model

Figure 7: Comparison model

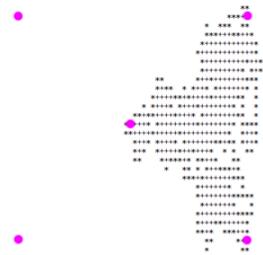
0 cycles



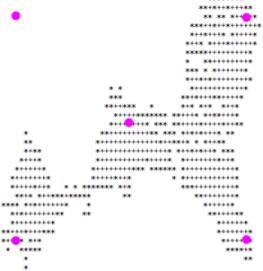
6000 cycles



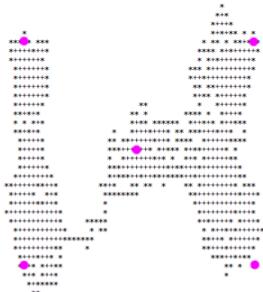
12000 cycles



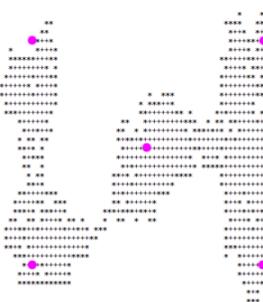
18000 cycles



24000 cycles



33000 cycles



Equivalent migration pattern of the simulated Physarum colony, with the same initialisation

6 Discussion

6.1 Chemotaxis

Chemotaxis has been shown to scale linearly over a regularised distribution of nodes (fig. 3a), with a linear rate of node-discovery that mimics that of a real *Physarum* sample (figs. 6, 7).

The difference in the rate of node discovery between the random and non random initialisations shown in fig. 3 demonstrates an issue with maintaining this scalability over different morphologies. The shallower gradient in 3b shows a ‘foraging penalty’: for ‘clumped’ initialisations of nodes far from the plasmodium, the chemoattractant signal is initially barely differentiable, making the main cost of the simulation the time taken to migrate toward a source of food. As the number of nodes increases, the likelihood of the far-off initialisation decreases, and the number of cycles required tends toward the value in fig. 3a.

The exploratory dendritic behaviour observed in fig. 6 is not emulated in this model, and it is this that allows real *Physarum* adaptability in searching for unknown food sources [1, 6, 10]. In a future project, it might be interesting to implement an extra function (such as the growth models in [21] or [6]) in combination with this model, to give a more realistic initial ‘searching’ behaviour.

6.1.1 Emergence

In order for the model of chemotaxis to be considered emergent, different behaviour must be observed for simulations initialised with the same conditions. [22] This is demonstrated in fig. 8, which shows four different simulations for the model described above, initialised with the same configuration, forming distinct morphologies.

6.2 Exploration-Exploitation

Fig. 4 shows that the self-optimising model is successful at reducing the average inter-nodal path length for a regularised configuration. However, both models contain some degree of self-stabilisation, observed in the fast decay of initial path length. The fluctuations in average path length in 4b are observed as more nodes added to the network, characterised by an initial rise followed by a smoothing-off as the optimisation gradually accommodates more nodes. In comparison to the real *Physarum*, this model falls short of the more dynamic self-strengthening and network minimisation behaviour observed in models by Tero et. al, and Houbraken et. al [20, 19, 7], which produce a much more accurate morphology.

Fig. 5 shows that the self-optimising model has poorer scalability over a large number of nodes than the initial model, with the number of cycles required to find all nodes greater on average, and increasing by a greater amount each time. The self-optimising behaviour thus engenders a trade-off between exploiting the already-found local network (minimising path length), and navigating towards

Figure 8: Visualisation of emergent behaviour



State of four different simulations after 30,000 cycles over a regularised nodal pattern, initial seed of 70 and 4-step growth phase

more potential food sources. Exploration-exploitation trade-offs in *Physarum* are also discussed in [6], where they form the basis for a discussion of scale-invariance in flocking networks.

6.3 Conclusions

Physarum networks exhibit a number of attractive properties in the field of decentralised network construction. In particular, scalable foraging, adaptive and self-optimising behaviour characterise the ‘morphological intelligence’ that underpins the computational fascination with *Physarum*.

This model demonstrates chemotactic network-forming behaviour as a cellular automaton, and a novel exploration of potential self-strengthening behaviour. In considering chemotaxis as a response to a $\frac{1}{r^2}$ signal transmission, it is possible to see how this algorithm might be used to establish clustering behaviour in a network of real agents. However, in order to implement a robust method of network formation based on this model, a more adaptable ‘searching’ behaviour would need to be established for ‘out-of-range’ agents.

The exploration-exploitation trade-off shows the potential for adaptive behaviour under different environments, and the emergent behaviour demonstrated by this model reflects the organic nature of the model. However, this model falls somewhat short of real *Physarum* morphology – better achieved in the original algorithm presented in [5] – a reminder that even small changes to the rule-set of a cellular automaton can produce entirely different morphologies [11, 22].

Although this model leaves room for improvement, it demonstrates successfully an emergent, cellular algorithm for network formation in free space. In combination with a more developed growth mechanism, this model has potential applications as a grounded and adaptive model of self-organising network formation.

Appendix A

The code for the simulations described in this report was written in C++, and is attached with compilation instructions in the files *slime_original.cpp* (initial model) and *slime_updated.cpp* (self-optimising model).

Acknowledgements

With thanks to Gary Zhexi Zhang for the use of his *Physarum* colony.

References

- [1] Adamatzky A. et. al, 2013, "*On Creativity of Slime Mould*", International Journal of General Systems, Vol. 42(5), 441-457
- [2] Adamatzky A. DeLacey-Costello B., Shirakawa T., 2008, "*Growing Spanning Trees in Plasmodium Machines*", International Journal of Systems Cybernetics, Vol. 37, 258-264
- [3] Fisher J., Henzinger T., 2007, "*Executable cell biology*", Nature Biotechnology, Vol. 25(11), 1239-1249
- [4] Gardner M., 1970, "*Mathematical Games: The fantastic combinations of John Conway's new solitaire game life*", Scientific American
- [5] Gunji Y. et al, 2008, "*Minimal model of a cell connecting amoebic motion and adaptive transport networks*", Journal of Theoretical Biology, Vol. 253, 659667
- [6] Gunji Y. et. al, 2015, "*An adaptive and robust biological network based on the vacant-particle transportation model*", Journal of Theoretical Biology, Vol. 272, 187-200
- [7] Houbraken M. et al, 2013, "*Fault tolerant network design inspired by Physarum polycephalum*", Natural Computing, Vol. 12(2), 277-289
- [8] <https://ig-nobel.silk.co/page/Toshiyuki-Nakagaki>, accessed 08/04/2017
- [9] Jones J. 2015 "*Mechanisms involving parallel computation in a model of Physarum Polycephalum transport networks*", Parallel Processing Letters, Vol. 25(1), 1540004
- [10] Jones J. 2016, "*Applications of Multi-Agent Slime Mould Computing*", International Journal of Parallel, Emergent and Distributed Systems, Vol. 31(5), 420-449
- [11] Jones J., 2010, "*Characteristics of Pattern Formation and Evolution in Approximations of Physarum Transport Networks*", Artificial Life, Vol. 16, 127-153
- [12] Nakagaki T., Yamada H., Hara M., 2001 "*Smart network solutions in an amoeboid organism*", Biophysical Chemistry, Vol. 107(1), 1-5
- [13] Pfeiffer, R., 1996, "*Symbols, patterns and behaviour: towards a new understanding of intelligence*", Proc. Japanese Conference on Artificial Intelligence, 1-15
- [14] Piccinini, G., 2015, "*Physical Computation: A Mechanistic Account*", Oxford University Press
- [15] Reid C. et. al, "*Decision making without a brain: how an amoeboid organism solves the two-armed bandit*", Journal of the Royal Society, Interface 13: 20160030, <http://dx.doi.org/10.1098/rsif.2016.0030>

- [16] Reynolds, C., 1987, "*Flocks, Herds, and Schools: A Distributed Behavioural Model*", Computer Graphics, Vol. 21(4), 25-34
- [17] Schulman A., 2015, "*Reversible logic gates on Physarum Polycephalum*", AIP Conference Proceedings 1648(1)
- [18] Song Y., Liu L., Ma H., 2012, "*A Physarum-Inspired Algorithm for Minimal Exposure Problem in Wireless Sensor Networks*", IEEE Wireless Communications and Networking Conference 04/2012, 2151-2156
- [19] Tero A., Kobayashi R., Nakagaki T., 2007, "*A mathematical model for adaptive transport network in path finding by a true slime mould*", Journal of Theoretical Biology, Vol. 244, 553-564
- [20] Tero A. et. al, 2010, "*Rules for Biologically Inspired Adaptive Network Design*", Science 327, 439-442, DOI: 10.1126/science.1177894
- [21] Tsompanas M-A., 2015, "*A cellular automata bioinspired algorithm designing data trees in wireless sensor networks*", International Journal of Distributed Sensor Networks, Vol. 2015, 2-12
- [22] Wolfram, 1984, *Computability theory of Cellular Automata*, Communications in Mathematical Physics 96, Springer-Verlag, 15-37