



University of the
West of England

Jones, J. (2011) Influences on the formation and evolution of *Physarum polycephalum* inspired emergent transport networks. *Natural Computing*, 10 (4). pp. 1345-1369. ISSN 1567-7818

We recommend you cite the published version.

The publisher's URL is <http://dx.doi.org/10.1007/s11047-010-9223-z>

Refereed: Yes

(no note)

Disclaimer

UWE has obtained warranties from all depositors as to their title in the material deposited and as to their right to deposit such material.

UWE makes no representation or warranties of commercial utility, title, or fitness for a particular purpose or any other warranty, express or implied in respect of any material deposited.

UWE makes no representation that the use of the materials will not infringe any patent, copyright, trademark or other property or proprietary rights.

UWE accepts no liability for any infringement of intellectual property rights in any material deposited but will remove such material from public view pending investigation in the event of an allegation of any such infringement.

PLEASE SCROLL DOWN FOR TEXT.

Influences on the formation and evolution of Physarum polycephalum inspired emergent transport networks

Jeff Jones

Natural Computing
An International Journal

ISSN 1567-7818
Volume 10
Number 4

Nat Comput (2011) 10:1345-1369
DOI 10.1007/s11047-010-9223-z



Your article is protected by copyright and all rights are held exclusively by Springer Science+Business Media B.V.. This e-offprint is for personal use only and shall not be self-archived in electronic repositories. If you wish to self-archive your work, please use the accepted author's version for posting to your own website or your institution's repository. You may further deposit the accepted author's version on a funder's repository at a funder's request, provided it is not made publicly available until 12 months after publication.

Influences on the formation and evolution of *Physarum polycephalum* inspired emergent transport networks

Jeff Jones

Published online: 30 October 2010
© Springer Science+Business Media B.V. 2010

Abstract The single-celled organism *Physarum polycephalum* efficiently constructs and minimises dynamical nutrient transport networks resembling proximity graphs in the Toussaint hierarchy. We present a particle model which collectively approximates the behaviour of *Physarum*. We demonstrate spontaneous transport network formation and complex network evolution using the model and show that the model collectively exhibits quasi-physical emergent properties, allowing it to be considered as a virtual computing material. This material is used as an unconventional method to approximate spatially represented geometry problems by representing network nodes as nutrient sources. We demonstrate three different methods for the construction, evolution and minimisation of *Physarum*-like transport networks which approximate Steiner trees, relative neighbourhood graphs, convex hulls and concave hulls. We extend the model to adapt population size in response to nutrient availability and show how network evolution is dependent on relative node position (specifically inter-node angle), sensor scaling and nutrient concentration. We track network evolution using a real-time method to record transport network topology in response to global differences in nutrient concentration. We show how Steiner nodes are utilised at low nutrient concentrations whereas direct connections to nutrients are favoured when nutrient concentration is high. The results suggest that the foraging and minimising behaviour of *Physarum*-like transport networks reflect complex interplay between nutrient concentration, nutrient location, maximising foraging area coverage and minimising transport distance. The properties and behaviour of the synthetic virtual plasmodium may be useful in future physical instances of distributed unconventional computing devices, and may also provide clues to the generation of emergent computation behaviour by *Physarum*.

Keywords *Physarum polycephalum* · Transport networks · Emergent behaviour · Network minimisation · Optimisation

J. Jones (✉)
Centre for Unconventional Computing, University of the West of England,
Bristol BS16 1QY, UK
e-mail: jeff.jones@uwe.ac.uk

1 Introduction

The single-celled multinucleate myxomycete organism *Physarum polycephalum* has been the subject of recent research interest because the transport networks formed during its growth and adaptation to the environment exhibit complex patterning and adaptive, distributed behaviour. The organism has been used as a living computing substrate and is capable of a number of complex distributed computing tasks which appear to transcend the properties of its simple component parts. In this paper we present a population based particle model of *Physarum* which collectively approximates the complex formation and evolution of transport networks seen in the organism. We study the properties of, and influences upon, the synthetic emergent transport networks and describe methods in which the particle population may, like *Physarum*, be considered as a (virtual) substrate for spatially represented computing tasks.

Physarum polycephalum, or true slime mould, has a complex life cycle and the vegetative plasmodium stage—where repeated nuclear division forms a giant syncytium within a single membrane—exhibits extremely interesting adaptive pattern formation behaviour including distributed growth, movement, food foraging, nutrient transport, hazard avoidance, and network adaptation and maintenance. As *Physarum* has no nervous system, or indeed any specialised tissue types, the question arises as to how the organism achieves these tasks?

During the plasmodium stage the organism is usually visible to the naked eye, its amorphous cytoplasm showing an amoeboid appearance. The plasmodium is a syncytium of nuclei within a cytoplasm comprised of a complex gel/sol network. Oscillations in the thickness of the membrane spontaneously appear with approximately 2 min duration (Takagi and Ueda 2007). The spatial and temporal organisation of the oscillations has been shown to be extremely complex (Takamatsu 2006). The protoplasm moves backwards and forwards within the plasmodium in a characteristic manner known as shuttle-streaming. The plasmodium is able to sense local concentration gradients and the presence of nutrient gradients appears to alter the structure of external membrane areas. The softening of the outer membrane causes a flux of protoplasm towards the general direction of the gradient in response to internal pressure changes caused by the local thickness oscillations. The strong coupling between membrane contraction and streaming movement is caused by the incompressibility of the fluid requiring a constant volume—the weakening of the membrane appears to provide an outlet for the pressure.

The active zone of the plasmodium surges towards nearby food sources showing strong inertial movement, engulfing and digesting them. Although the growth patterns differ depending on nutrient availability and growth substrate properties (Adamatzky et al. 2008; Takamatsu et al. 2009), the growth front typically moves outwards with a fine mesh-like, almost uniform, connectivity at the actively growing region, with an adaptive coarsening of connectivity towards the origin. When the plasmodium has located and engulfed nearby food sources, protoplasmic veins appear within the plasmodium, connecting the food sources. The purpose of the veins is to transport protoplasm amongst the distributed extremes of the organism. The effect is to both maximise foraging area (during the foraging stage) and to minimise nutrient transport distance by the formation of the protoplasmic veins, whilst at the same time maintaining a fault tolerant network connectivity that is resilient to damage to any particular part (Nakagaki et al. 2004a).

The relative simplicity of the cell and the distributed nature of its control system make *Physarum* a suitable subject for research into distributed computation substrates. In recent years, there has been a wealth of research into its computational abilities, prompted by

Nakagaki et al. who initially reported the ability of *Physarum* to solve path planning problems (Nakagaki et al. 2000). Subsequent research has confirmed and broadened the range of abilities to spatial representations of various graph problems (Nakagaki et al. 2004b; Shirakawa et al. 2009; Adamatzky 2008b), combinatorial optimisation problems (Aono and Hara 2007), construction of logic gates (Tsuda et al. 2004) and logical machines (Adamatzky 2007), and as a means to achieve distributed robotic control (Tsuda et al. 2007) and robotic amoebic movement (Ishiguro et al. 2006).

A number of different models, and modelling approaches, have been used to try to either understand the complex behaviour of *Physarum*, or to generate similar behaviours computationally. Tero et al. have suggested that protoplasmic flux through the network veins may be cause the physical basis for evolution of the transport network: given flux through two paths, the shorter path will receive more sol flux. By generating an autocatalytic canalisation mechanism to reward veins with greater flux (by thickening/widening them) and to apply a cost to veins with less flux (these veins become thinner), shorter and thicker veins begin to predominate as the network evolves. This approach was used for the mathematical model of *Physarum* network behaviour to solve path planning problems (Tero et al. 2007). The starting point for the model of Tero et al. is a randomly connected protoplasmic tube network, surrounding a number of food sources (network nodes) which act as sources and sinks of flux. This corresponds to an area completely covered by the plasmodium before addition of food sources to initiate network minimisation. By beginning with a complete network this method, although successful in generating impressive solutions to network problems, sidesteps the issue of initial network formation. Gunji et al. (2008) introduced a cellular automaton (CA) model which considered both plasmodial growth and amoeboid movement. Their model placed importance on the transformation of hardness/softness at the membrane and the internal transport of material from the membrane resulting in movement and network adaptation. The model was also able to approximate instances of maze path planning and coarse approximations of the Steiner tree problem.

Takamatsu's hexagonal CA (Takamatsu et al. 2009) is an early stage model initially concerned with mimicking the growth patterns displayed under differing nutrient concentrations and substrate hardness. The patterns reflect experimental results well but do not (at least at this stage—oscillatory behaviour is in development) show adaptation as the plasmodium grows. Hickey and Noriega (2008) adapted a classical ant algorithm to modify a decision tree in their representation of *Physarum* behaviour in a simple path planning problem. Their algorithm (as with many implementations of ant algorithms) transformed the spatial area into a graph representation and provided broadly similar results to path optimisation by *Physarum*. Although not concerned with *Physarum* at the time, the growth of a plasmodium from a single node to create a spanning tree appears similar to the behaviour of a neural growth algorithm by Adamatzky (1991).

Current attempts at modelling *Physarum* often compartmentalise different behaviours in an attempt to simplify the huge overall modelling task, for example providing different representations and mechanisms for growth, movement, internal oscillations and network adaptation. As yet there is no single model that completely encapsulates *Physarum* behaviour (and this is only considering the plasmodium stage of the life cycle). The different modelling approaches also have different implementations—from the purely spatial CA models, to mathematical representations of flux canalisation, oscillatory behaviour (Kobayashi et al. 2006; Nakagaki et al. 1999), and path length.

In this paper we are particularly concerned with the ability of *Physarum* to construct and evolve transport networks. Nakagaki et al. (2004a) demonstrated that when a dense

plasmodium network is presented with multiple food sources, the plasmodium adapts its transport network to cover the food sources and minimise network length. The topology of the networks was most often a combination of the minimum Steiner tree and extra cyclic connections between the nodes which maintained connectivity in response to random tube disconnection. Adamatzky recently suggested that a growing *Physarum* plasmodium approximates the increased connectivity of the Toussaint hierarchy of proximity graphs (Toussaint 1980; Jaromczyk and Toussaint 1992). Proximity graphs are graphs connecting a set of points whose connectivity is determined by particular definitions of neighbourhood and distance. For example in the construction of the relative neighbourhood graph (RNG) two points p and q are connected only when there is not a third point r that is closer to both p and q than the distance between p and q . Each subsequent member of the Toussaint hierarchy of proximity graphs contains the links of the graphs earlier in the hierarchy, adding extra links to satisfy the different neighbourhood definitions. *Physarum* was shown to initially create a spanning tree (single node inoculation) and increased its connectivity to span more connected types of proximity graph (Adamatzky 2008a), most closely approximating RNG. When inoculated at all nodes simultaneously the plasmodium approximated higher connectivity members of the proximity graph family, the Gabriel Graph (GG) and Delaunay Triangulation (DTN). Adamatzky suggested that the growth front of the plasmodium under different environmental conditions corresponded to the neighbourhood definition of different proximity graphs (for example the lune neighbourhood of RNG or the circular neighbourhood of GG). Therefore, nutrient poor environments would be expected to induce more sparsely connected networks than nutrient rich environments (for example, a weak stimulus from only one direction would present a conical chemoattractant profile, whereas multiple strong stimuli would present a radial chemoattractant profile).

We present a simple particle population model of *Physarum* which approximates the complex network formation and evolution seen in the plasmodium. The particle population collectively behaves as a virtual ‘material’ which can then be exploited for unconventional computing implementations of geometry problems. Section 2 introduces the basic particle model and discusses network formation and evolution under a variety of different environmental and parametric conditions. Section 3 considers the evolution of the networks when constrained by the projection of sources of diffusing chemoattractant (simulated food sources) and the adjustment of network configuration by changing the strength of chemoattractant projection. Section 4 illustrates examples of the particle approximation of proximity graphs in the Toussaint hierarchy and other related geometry problems. Section 5 presents an extension to the model to allow for dynamic adjustment of population size in response to environmental conditions. We use this method to investigate the role of relative node positions (specifically angles) and nutrient concentration. Using a real-time method to track network topology we examine how network evolution and configuration is affected by global changes in network node projection strength. A discussion of the computational resources, limitations and possible advantages of the approach is given in Sect. 6. Finally a summary of the emergent transport network approach is given in Sect. 7 along with a brief description of ongoing and future work.

2 A particle approximation of a virtual plasmodium

The model is based on simple particle behaviours that were used in (Jones 2010b) to generate emergent transport networks. The model extends previous work which explored

passive particle swarm approximations of reaction–diffusion (RD) computing. By passive we mean a particle population which responds to the diffusive environment but which does not alter the diffusive environment. It was shown that a passive swarm approach could generate solutions to spatially represented geometry problems (such as skeletonisation, path planning and Voronoi diagram approximation) with dynamically changing datasets (Jones 2008). This paper extends the swarm behaviour to include active approaches to RD computing—the swarm population not only senses but modifies the diffusive environment [for a comparison of the passive and active approaches see (Jones 2009)].

The model is computationally very simple, using distributed local sensory behaviour, but approximates some of the complex phenomena observed in *Physarum*. A single particle, and an aggregation of particles, is related to the *Physarum* plasmodium in the following way: We conceptualise and simplify the plasmodium syncytium as an aggregate of identical component units (formed by repeated nuclei division). Each ‘unit’ represents a hypothetical unit of gel/sol interaction. Gel refers to the relatively stiff sponge-like matrix composed of actin–myosin fibres and sol refers to the protoplasmic solution which flows within the matrix. In *Physarum* the resistance provided by the gel matrix to the flow of sol, and the spontaneous contraction of the actin–myosin fibres, results in changes in thickness of the plasmodium due to hydrostatic pressure and a rhythmic streaming flow of sol throughout the matrix. Ultimately a complex set of distributed oscillations (of sol flux and plasmodium thickness) is observed.

In the particle model the production and flow of sol is provided by the movement of the particles and the resistance of the gel matrix is provided by the overlapping sensory coupling and the particle–particle collisions (one can imagine a tightly crowded population within a room, with restricted movements). The collective position of the particle population represents the *structure* of the *Physarum* transport network, and the movement of particles represents the *flow* of protoplasmic sol within the network. The morphology of each particle is shown in Fig. 1. The particle ‘body’ occupies a single cell in a discrete lattice and each cell corresponds to a pixel in a digitised greyscale image. The greyscale values of the image correspond to the environment habitat of the population (specific grey levels representing empty space, boundaries, food sources and hazards, where appropriate). The particle has three sensors which sample the environment values at some distance away (SO—sensor offset distance, in pixels) from the particle. This offset sensor morphology generates significant sensory local coupling between the particle population (the sensory input of one particle can be affected by the actions of a nearby particle). The cohesion of this aggregate ‘crowd’ is ensured by the fact that there is mutual attraction to the stimuli deposited by the particle population. The population adopts autocrine chemotaxis

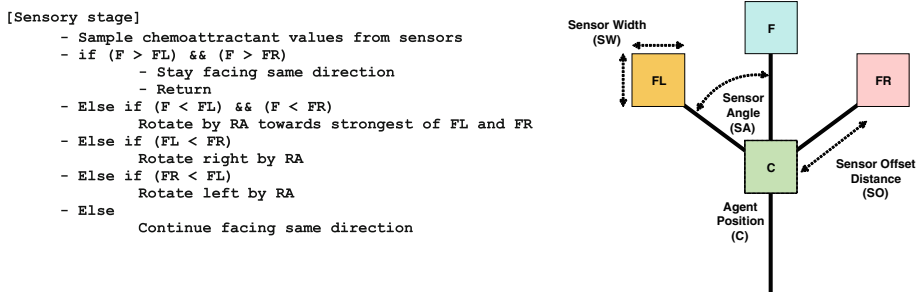


Fig. 1 Particle sensory behaviour and particle morphology

behaviour (meaning that particles both secrete and sense approximations of the same chemoattractant, so the actions of the particles can also affect their behaviour). One significant simplification with respect to the real organism is that both food sources and the changes in flux of internal protoplasm are represented by the same diffusing chemoattractant substance. The built in lag between the particle movement (a simplistic approximation of actin/myosin contraction movement) and the production and diffusion of protoplasmic sol flux results in complex emergent population behaviour. Collective movement, shape minimisation and internal oscillations of the plasmodium are emergent properties of the aggregate behaviour of the particle population and are not explicitly coded into the particle algorithm. Directional orientation and movement of the plasmodium is generated by coupling the emergent mass behaviour with chemoattractant to local food source stimuli.

In the sensory stage of the algorithm, the particles sense the concentration of stimuli in the chemoattractant map from three forward oriented sensors. At each scheduler step the particles orient themselves towards the strongest chemoattractant source by rotating left or right, depending on the source of the stimuli. Note that the particle behaviour is very simple and is explicitly forward biased—the particle does not contemplate its current position—there is an implicit emphasis on forward movement. There is significant interplay between the sensor arm angle (SA) and the rotation angle (RA) parameters. When the two parameters are equal (both 45°), the effect on network formation is to contract the initial network that emerges. If $RA < SA$ the contraction behaviour is increased. If, however, $RA > SA$ then spontaneous branching will appear during the network formation. The branching occurs because the larger rotation angle places a particle's sensors outside the area of the chemoattractant trail path that the particle was following. The RA and SA parameters can be used to tune the network behaviour towards minimisation or towards high connectivity. A parametric investigation of different RA vs. SA values has recently been completed and it was found that the patterning abilities of the networks is equivalent to classical models of reaction–diffusion patterning processes, including spotted, striped, labyrinthine and reticular patterning (Jones 2010a).

After the sensory stage is completed the scheduler executes the motor stage where each particle attempts to move forward in its current direction (represented by an internal state from 0 to 360°). If the new site is not occupied, the particle moves to the new site and deposits chemoattractant trail onto the chemoattractant map at the new location (arbitrary units). If the particle cannot move forwards successfully then no deposition is made to the chemoattractant map. Inertial movement for each particle is also possible, resulting in the spontaneous emergence of spatial oscillations within the population. The inertial behaviour provides scope for amoebic movement and external control but for space reasons is not considered in this paper. The particle population is updated in a random order for both the sensory and motor stages to avoid any influence from sequential positional updates. Each sensory and motor update, combined with environmental diffusion and degradation of chemoattractant is considered as one scheduler step.

For the particle approximation of *Physarum* plasmodium synthetic environments were created for the model organism. A population (%p, where p is the percentage of environment size in pixels) of mobile particles is created. A two-dimensional discrete map represents the environment configuration. To provide a synthetic representation of *Physarum*'s preferred nutrients (for example, oat flakes) we use virtual chemoattractant food source stimuli whose positions are stored in the problem configuration map. To implement the diffusion of the synthetic chemoattractant the stimuli are projected to a coupled map (the *chemoattractant* map) at every step of the scheduler. The strength of the

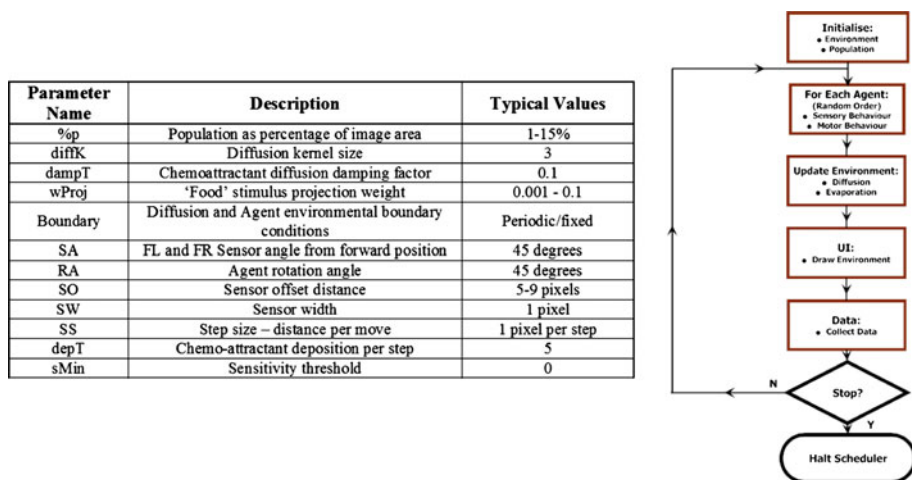


Fig. 2 Environment and particle parameters and scheduler operation

chemoattractant stimuli (in arbitrary units) can be adjusted with a weighting parameter. Larger values of this weight parameter reflect stronger (and more attractive) food sources and smaller values reflect weaker (less attractive) sources. The chemoattractant stimuli diffuse throughout the environment by means of a simple 3×3 mean filter kernel. The diffusing chemoattractant values may be damped to mimic the spatial decay of stimulus strength in the environment and thus maintain a relatively steep diffusion gradient away from the food source. The diffusion gradient corresponds to the strength of the nutrient substrate of the plasmodium's environment (for example the different growth patterns seen in damp filter paper and agar substrates). Differences in the stimulus strength and stimulus area (the *size* of food source), affect both the steepness, and propagation distance of the diffusion gradient and affect the growth patterns of the synthetic plasmodium. The experiments on the model plasmodium were designed and configured to reflect 'wet' experiments reported in experiments with the organism by Adamatzky (2008a). The food sources correspond to the oat flake positions used in the wet experiments and are indicated as small dark spots. The plasmodium/particle flux is indicated as rough textured areas where darker areas correspond to greater chemoattractant concentration. The behaviour of the scheduler during an experimental run is shown in Fig. 2 along with a list of model parameters and their typical values.

2.1 Formation and evolution of dynamic transport networks

Without the presence of pre-existing chemoattractant stimuli, the particle population quickly self-assembles into network structures. The networks are dynamical (composed of the changing movement of particles) and evolve over time in a complex manner (Fig. 3). The networks paths undergo substantial modification—the initially fine meshwork evolves into patterns of paths and lacunae. The thickness of the paths, and scale of the network features, is related to the sensor offset (SO) parameter. Larger values of SO generate thicker, coarser paths with larger features (not shown). The smaller lacunae gradually close by sphincter-like contraction: two meeting network paths merge together in a zip-like manner and, because the shorter path results in greater flow, the circular paths close and the

network adapts to the new configurations. With the non-inertial motor behaviour and particle settings of SA 22.5°, RA 45° the closure of lacunae continues until new branch-like processes sprout from existing paths (final two images in Fig. 3). These branches cross pre-existing lacunae and merge with the network flow on the other side by anastomosis. The changing network configurations continue indefinitely (although occasionally with temporary periods of stability). When both RA and SA are equal to 45°, the spontaneous branching behaviour does not occur and the closure of lacunae continues until a minimal surface is reached. With periodic boundary conditions (diffusion continues across the boundary of the lattice to the opposite edge) this surface connects all sides of the two-dimensional (toroidal) environment which, when tiled, approximates a hexagonal pattern, known to be the minimal connectivity in two-dimensions (Hales 2001) (Fig. 4).

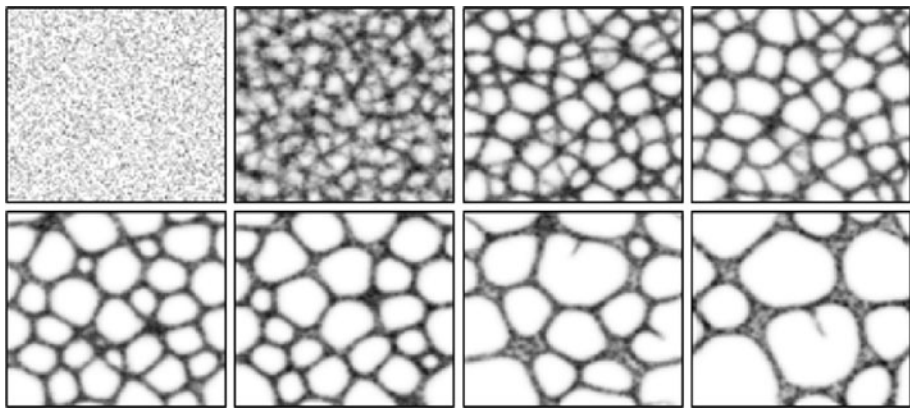


Fig. 3 Spontaneous formation and dynamical evolution of transport networks. Lattice 200×200 , %p15, RA45°, SA22.5°, SO 9. Images taken at: 2, 22, 99, 175, 367, 512, 1740 and 4151 scheduler steps

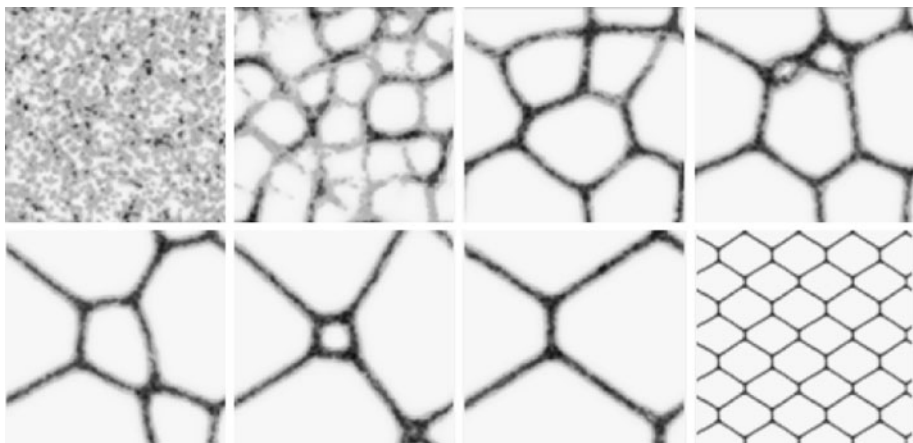


Fig. 4 Stable minimising network with periodic boundary conditions approximates hexagonal tiling. *Left to right* evolution and condensation of network from initial random state. Final figure (*bottom right*) is a tiled version of penultimate figure, indicating honeycomb tiled pattern

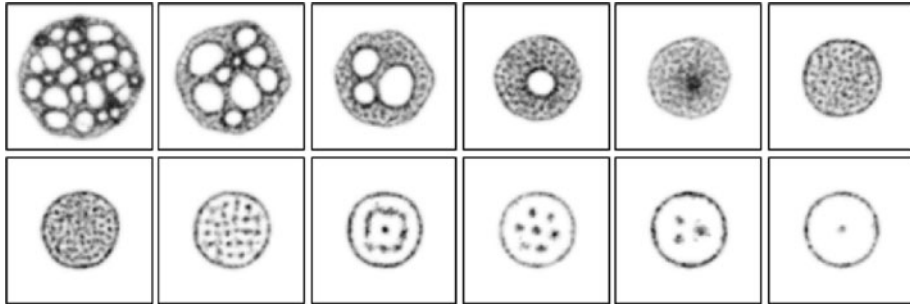


Fig. 5 Formation of plasmodial sheet structure from condensation of fixed boundary emergent transport networks. *Top row, left to right* Network evolution over time: %p = 20 particle trails, RA45°, SA 45°, SO 9. *Bottom row, left to right* Stable vacancy island patterns (dark spots) at SO: 9, 13, 19, 23, 28 and 38

When boundary conditions are fixed, i.e. when the diffusing chemoattractant is absorbed at the edges of the two-dimensional lattice, the network condensation continues until a solid mass is formed. This mass evolves into a minimal circular shape (Fig. 5). When a larger population is used the mass takes the form of a sheet-like structure, resembling the giant plasmodium of *Physarum*. The sheet mostly consists of immobile particles (particle that cannot deposit chemoattractant as there is little free movement) and most of the movement occurs at the border of the sheet. There are, however, small regularly spaced areas within the sheet where movement occurs. These vacancy ‘islands’ are small vacant spaces where particles may move. The islands are dissipative and mobile and appear to be related to the sensor offset scale of the particles. As the scale of the particle sensor coupling increases (SO parameter), the number of vacancy islands falls.

2.2 Mechanisms and motifs of network self-assembly

When the population is initialised, the diffusing chemoattractant deposited by the initially random movement attracts nearby particles. The local activation of nearby particles creates an autocatalytic amplification of trail on the path which the particles move along. The positive feedback nature of the attraction also results in a negative feedback response: areas that have lower population densities have less chemoattractant and, therefore, are less likely to attract particles. The result is that the paths become self-reinforcing.

This only explains the initial network formation, however. The dynamical evolution of the network (closing of lacunae and shortening of paths) is due to the fact that particle–particle collisions result in random changes in orientation. Instead of being deadlock when particles collide, the particles can easily *slide past* each other. The inner areas of a cyclic connection are not encumbered by the spokes of the network paths feeding the cycle. Chemoattractant flux is also naturally greater in the inside of a cycle due to the smaller diameter. As particles move inwards towards areas of greater flux the radius becomes even smaller and flow increases until finally the cycle is closed. The complex behaviour of the networks is also due to the fact that there are a constant number of particles within the network. When one cycle closes, the mass of particles which was contained within the cycle is now free to migrate to other parts of the network. The introduction of new particle flux into different parts of the network affects the long term network structure. Also of note is the facts that the flow along the networks is bi-directional—different particles migrate across network paths in both directions simultaneously. Schematic illustrations of the

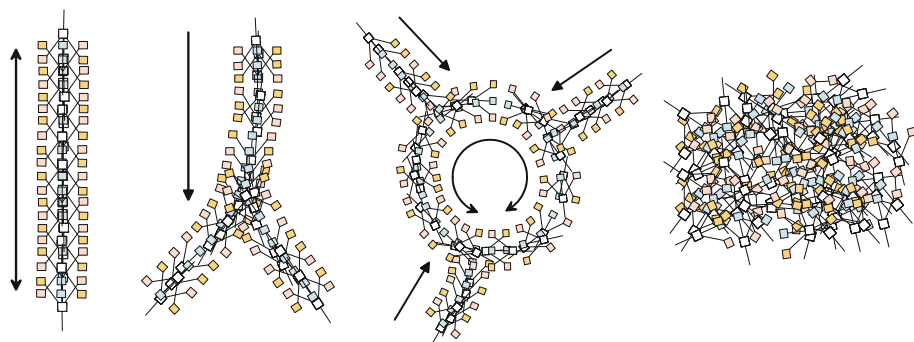


Fig. 6 Illustration of particle flow motifs in self-assembled networks. *Left to right* bi-directional network path, minimising ‘zipper’ path motif, closing of lacuna cycle, plasmodial sheet mesh. Note that the zipper is closed in the downwards direction even with bi-directional flow

network flows are shown in Fig. 6. The particles appear to adopt a polar alignment with the ‘head’ of the particle facing the inside of the network paths and plasmodial sheets. The polar alignment of the particles appears to mimic, at least superficially, the self-assembly of lipid molecules into lipid vesicles. The self-organising features of the particle networks may provide a useful model of recent experiments by Lobovkina et al. (2008) in the self-assembly and automatic reconfiguration in lipid nanotube networks. As with the lipid nanotube networks the collective particle behaviour appears to approximate the evolution of soap films and adhere to Plateau’s laws for two-dimensional foams in terms of branching junctions (only ‘Y’ shaped junctions persist) and branch angles (120° angular symmetry at stabilised junctions).

3 Network evolution around chemoattractant sources

The mechanisms that guide self-assembly and evolution of the networks may be harnessed for computational means by providing external sources of chemoattractant. These sources attract and anchor the condensing networks. Three methods may be utilised which loosely correspond to different methods of inoculation of the *Physarum* plasmodium. These methods are filamentous condensation (FC: formation of a spatially distributed network from random seed points), filamentous foraging (FF: network initialised at food source nodes), and plasmodial shrinkage (PS: network evolution by shrinkage of a plasmodial sheet). After using simple exemplar network configurations and discussing methods of altering network structure we adopt the food source node positions used by Adamatzky (2008a) who found that *Physarum* networks followed the upper part of the Toussaint hierarchy of proximity graphs when inoculated at all nodes simultaneously. Given that *Physarum* plasmodium can perform network optimisation problems; does the synthetic virtual plasmodium also possess such capabilities?

3.1 Filamentous condensation method (FC)

A relatively small population (%2) is initialised at random locations and orientations. A transport network emerges and the network becomes ‘snagged’ on the food sources due to attraction to the chemoattractant gradients emitted from the nodes. The evolution of the network continues with the constraints of the food sources affecting the topology. As the

network seeks to maximise flow, cycles within the network are closed and curved paths tightened. The final configuration of the network shows the minimal transport path length, characteristic of the Steiner tree (Fig. 7, top row).

3.2 Filamentous foraging method (FF)

In this method the population is initialised from the food source node positions. A fixed population size was used (a method to grow and shrink the population size in response to environmental conditions is in development) and the particles were released onto the food sources gradually as space became available. The network patterns formed depended on the SA parameter (RA was fixed at 45°). When SA was low ($11.25\text{--}22.5^\circ$) fine branching networks were formed with constantly changing topologies. These network shapes approximated the Delaunay Triangulation graph (Fig. 7, middle row, first two columns). When SA was 45° , minimising networks were formed, towards Steiner tree structures (remaining columns).

3.3 Plasmodial shrinkage method (PS)

Instead of seeding and condensing a network around the chemoattractant stimuli points it is possible to use the stimuli to deform a pre-existing plasmodial sheet. The approach used is based on the behaviour of the *Physarum* plasmodium which, after initial coverage of the environment, shrinks to form networks of transport veins connecting the food sources. The plasmodial shrinkage method populates the environment with a very large number of particles ($\%p = 50$ in a 300×300 lattice: 45,000 particles) to ensure that the network forms a solid sheet structure. As the experiment progresses particles, chosen at random, are removed from the environment. Particles are never re-admitted to the environment once removed. The probability of particle removal was set to 0.00025 for each particle at every scheduler step. At early stages of the experiment (when the population was high) the removal rate is relatively high. As the population size decreased so did the rate of removal.

As the particles are randomly removed, the sheet dynamically deforms and reconfigures itself (Fig. 7 bottom row), influenced by the strong gradients of the pre-pattern stimuli

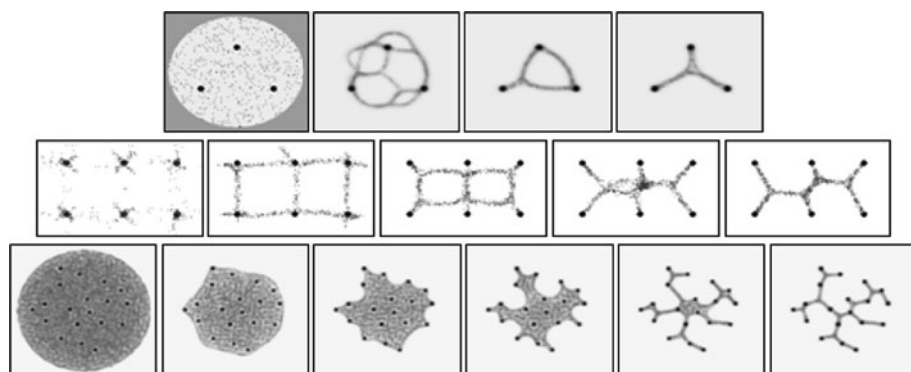


Fig. 7 Three methods of initialising virtual plasmodium for network minimisation problems. *Top row* FC method (lattice 200×200 , RA 45° , SA 45° , SO9, $\%p2$). *Middle row* FF method (lattice 254×170 , RA 45° , SA Initially 22.5° then 45° , SO 9, $\%p2$). *Bottom row* PS method (lattice 300×300 , RA 45° , SA 45° , SO 9, $\%p50$)

(corresponding to *Physarum* food sources). The relatively slow rate of particle removal ensures that the tension in the sheet is uniformly distributed and that particle flux can adapt in time to the changing environment (faster rates of removal results in the spontaneous formation of cyclic areas—the sheet deformation is unbalanced resulting in a tearing of the sheet and holes being formed). The sheet maintains a minimal area as it shrinks and the final result approximates a minimum spanning tree.

3.4 Adjusting network evolution using tension mechanisms

In the three methods discussed above, the evolution of the networks is affected by the attraction of the particles to each other, and to the network nodes (chemoattractant sources). If the concentration of chemoattractant at the network nodes is less than the amount secreted by the particles, the nodes have no influence on the network evolution (which continues to a circular mass). If the concentration is much greater at the nodes, the particles preferentially seek the node sources and the emergent tension forces (especially in the plasmodial sheet) become more apparent. This method can be also be used to ‘unzip’ Steiner points in network paths (Fig. 8). If the force is too large the network paths may tear apart as the particles migrate to the nodes. Reducing the tension (by decreasing node concentration) causes the Steiner points to re-form as the network relaxes, and reducing network tension may also be used to close network cycles otherwise trapped by an external arrangement of nodes.

4 Spatially represented geometry problems and the virtual plasmodium

Figure 9 shows results utilising the minimisation behaviour of the virtual plasmodium to tackle network optimisation problems. Fixed population sizes were used and the population behaviour was initialised with highly foraging (branching) behaviour ($SA < 45^\circ$ and $RA = 45^\circ$). This reduces differences caused by random variations in particle initialisation positions. To induce minimisation behaviour SA was increased to 45° , stopping new network branching and minimising the network by surface tension effects. It should be noted that the network configurations shown in the image snapshots pertaining to the traversal of the Toussaint hierarchy are transient in nature and part of a continuum of minimisation evolution—the images were selected by the author manually as a visual aid to relate to the movement down the hierarchy. During the foraging stage, the networks are

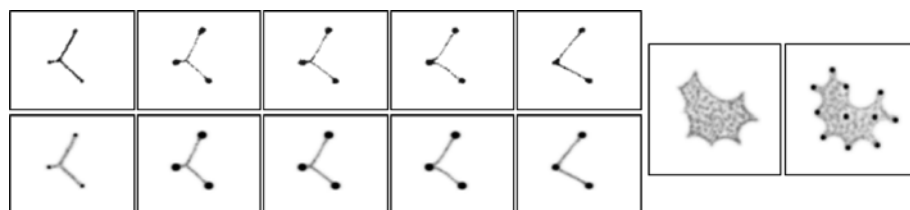


Fig. 8 Increasing node chemoattractant concentration increases tension effects and alters network configuration. *Left* A three node network with Steiner point adapts to remove Steiner point when node chemoattractant concentration is increased. Particle positions on *top row*, chemoattractant nodes and trail paths on *bottom row*. *Right* Plasmodial sheet under low tension (*left*) adapts its shape to increase in tension (*right*) caused by increase in node chemoattractant concentration

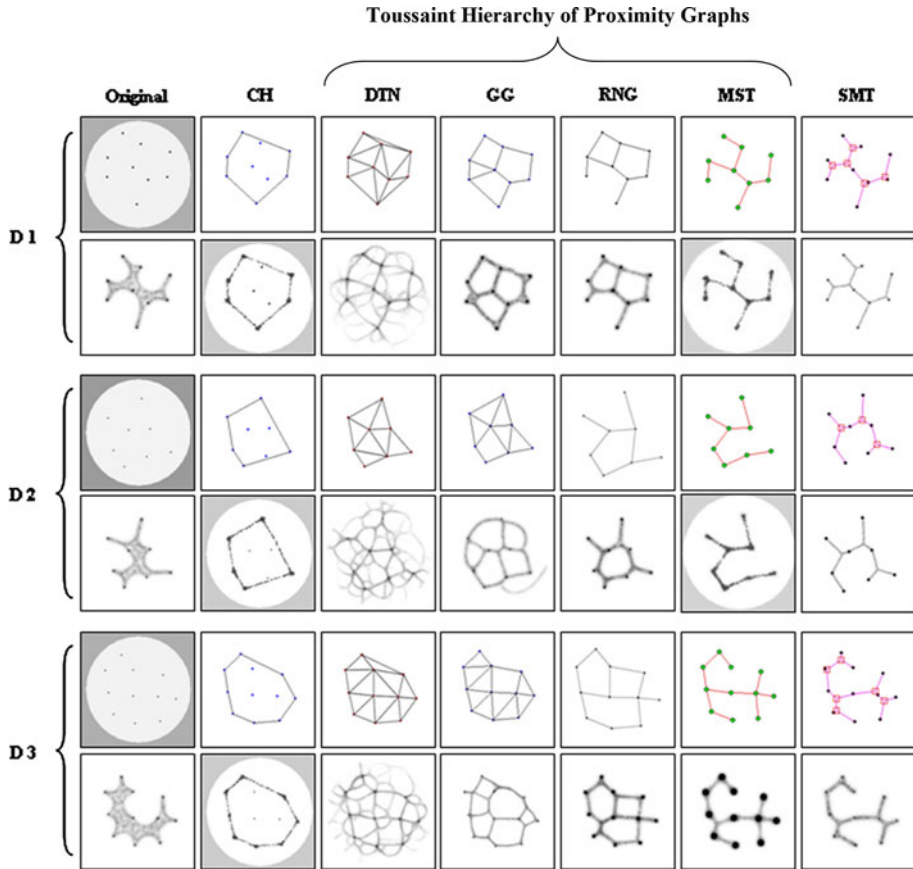


Fig. 9 Applying virtual plasmodium to network minimisation problems and illustrating the traversal down the Toussaint hierarchy. Three separate datasets shown: D1, D2 and D3. *Top Row of Each Dataset* Original node points, convex hull (CH), Delaunay Triangulation (DTN), Gabriel Graph (GG), Relative Neighbourhood Graph (RNG), Minimum Spanning Tree (MST), Steiner Minimum Tree (SMT). *Bottom Row of Each Dataset* Example snapshots from collective approximation of: Concave Hull (see text), Convex Hull, DTN, GG, RNG, MST, SMT

initially very chaotic with very frequent changes in topology. Some of the network paths during this stage are very thin indeed and do not persist for a long time, whereas others persist and grow thicker over time. This behaviour was also observed by Adamatzky in the construction of the higher (RNG > GG > DTN) stages of the hierarchy. Also of note is the fact that the changes in network topology were greater in the areas near the perimeter of the set of points, the interior of the network (with a greater number of nearby nodes) was more stable.

As the minimisation progresses the number of cyclic connections (and network length) is reduced and the synthetic plasmodium adapts to the changing configuration of the network until a final stable configuration is reached. All three methods were assessed for each dataset, each method showing differences in network minimisation. For the D1 dataset when using both the FF and FC approaches, the final structure was typically the MST, and for D2 and D3 the final network structure typically had one and two cycles,

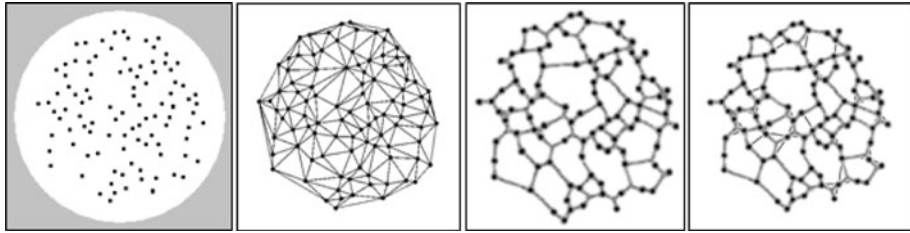


Fig. 10 Complex networks are minimised by the virtual plasmodium to approximate the relative neighbourhood graph. *Left to right* Source nodes, initial network configuration (DTN), final network configuration, network overlaid to RNG

respectively. When using the PS method the final structure was a tree, typically approximating variations of the SMT (which could be further ‘tightened’ to obtain the MST).

For networks with a large number of nodes it would be expected that the minimisation would not naturally continue to a tree structure because the arrangement of large number of nodes would prevent cyclic closure across large distances. To assess how far down the Toussaint hierarchy the minimisation would naturally progress, a synthetic plasmodium network was initialised with a large number of nodes in the topology of the DTN. The network began to minimise by closing cyclic areas and the final state was closest in appearance to the RNG (Fig. 10), supporting the hypothesis that network minimisation in biological transport networks is a compromise between network length and network resilience.

The convex hull approximation (the smallest convex polygon enclosing a set of points where all points are on the boundary or interior of the polygon) was achieved by initialising particles in a circular ring outside the datasets and applying a strong tension (via high node chemoattractant deposition) to the network. As the network contracts it is ‘snagged’ at the outer nodes and the stable pattern reveals the convex hull. The concave hull is related to the convex hull and is commonly defined as the minimum region (or *footprint*, Galton and Duckham 2006) occupied by a set of points, which cannot, in some cases, be represented correctly by the convex hull (Duckham et al. 2008) (for example, a set of points arranged to form the capital letter ‘C’ would not be correctly represented by the convex hull because the gap in the letter would be closed). The virtual plasmodium approximates the concave hull when all of the particles are within the region of the data source nodes. The approximation is achieved by the plasmodial shrinkage method and the shape of the concave hull approximation can be adjusted by changing the node tension and/or particle sensory scale length.

5 Adaptive population size and transport network analysis

A fixed population size limits the functionality of the model. A sufficiently large population must be specified to allow for the discovery and connection of all possible food sources yet, if the population is too large, the minimised final transport network will have paths which are very thick. Although, it is possible to estimate the required population size based on the number of nodes, such a calculation is rather artificial since it relies on prior global knowledge of the environment. Furthermore, such a scheme could not cope with the addition or removal (or exhaustion of) food sources.

5.1 Environment adaptation: population growth and shrinkage

Growth and adaptation of the particle population is implemented using a simple method based upon local measures of space availability (growth) and overcrowding (adaptation, or shrinkage, by population reduction). This is undoubtedly a gross simplification of the complex factors involved in growth and adaptation of the real organism (such as metabolic influences, nutrient concentration, waste concentration, slime capsule coverage, bacterial contamination). However, the simplification renders the population growth and adaptation more computationally tractable and the specific parameters governing growth and shrinkage are at least loosely based upon real environmental constraints. Growth and shrinkage states are iterated separately for each particle and the results for each particle are indicated by tagging Boolean values to the particles. The growth and shrinkage tests were executed every three scheduler steps and the method employed is specified as follows.

If there are between 1 and 10 particles in a 9×9 neighbourhood centred at the location of a particle, and the particle has moved forwards successfully, the particle attempts to divide into two if there is an empty location in the immediate neighbourhood surrounding the particle. For the shrinkage behaviour we assess a 5×5 neighbourhood around the particle. If there are between 0 and 20 particles in region the particle survives, otherwise it is annihilated.

Growth of the population is implicitly linked to movement (since a particle can only divide if movement is successful). This ensures that particle growth is more likely to be initiated at the border of a mass of particles and when movement is stimulated by local nutrient availability. The shrinkage of the population when overcrowding occurs ensures that network paths between nodes remain thin. The adaptive population size also prevents 'snapping' of the network when apparent network tension is increased by increasing the strength of chemoattractant projection at nodes. If a break is caused in the transport network due to increased tension, new particles are created to repair the break. Conversely, when tension is reduced (by decreasing projection strength) the population size adapts by shrinking if local crowding is too high.

5.2 Factors affecting network evolution

The gallery of results in Fig. 9 show that the transport networks traverse the Toussaint hierarchy under a variety of foraging conditions and node projection strengths. Even though the number of nodes in each of the three data sets ranged from only 8 to 11, the actual network patterns were very different, suggesting that node position and, consequently, inter-node angles are important factors in guiding network evolution. To examine the factors affecting evolution we assessed the simplest case of a three node network where a central node is surrounded by two outer nodes at equal distance and equal angles. A small population (100) was initialised on a horizontal line spanning the three nodes. Initially the angle between the two outer nodes was 180° and this angle was systematically decreased at regular intervals (one degree every 200 scheduler steps), pivoting the outer nodes around the central nodes. The population size was dynamically adapted using the above method to maintain network paths of uniform thickness. When a critical angle was reached the two outer network paths touched near the vicinity of the central node and merged. The mutual attraction of the flows pulled the network away from the central node and the direct connection to the central node was broken by the characteristic 'zipping' motif. The zipping behaviour continued until the three competing network flows stabilised at a Steiner point. The critical angle was measured at different node projection strengths of 0.005, 0.05,

0.5 and 5, and at different sensor offset scales of 5, 9 and 13 pixels and a summary of the results is shown in Fig. 11.

The results summarised in the chart indicate that both node projection weight and sensor offset distance affect the size of the critical angle. Increasing the sensor offset distance resulted in the critical angle occurring at larger angles than with smaller offset distance. This is because a larger sensor offset distance results in thicker network paths and these paths come into proximity with each other at relatively larger network angles. The effect of node projection weight is more pronounced. Smaller weights do not exert as great an influence on the network paths as do larger weights (indeed at SO 5 and weight 0.005 the attraction of the nodes was not strong enough to constrain the paths reliably, resulting in serpentine foraging network paths, and no information on the critical angle could be obtained). At high projection weights the influence of the nodes resulted in a much smaller critical angle and also a wider region of influence on nearby network paths. Very high

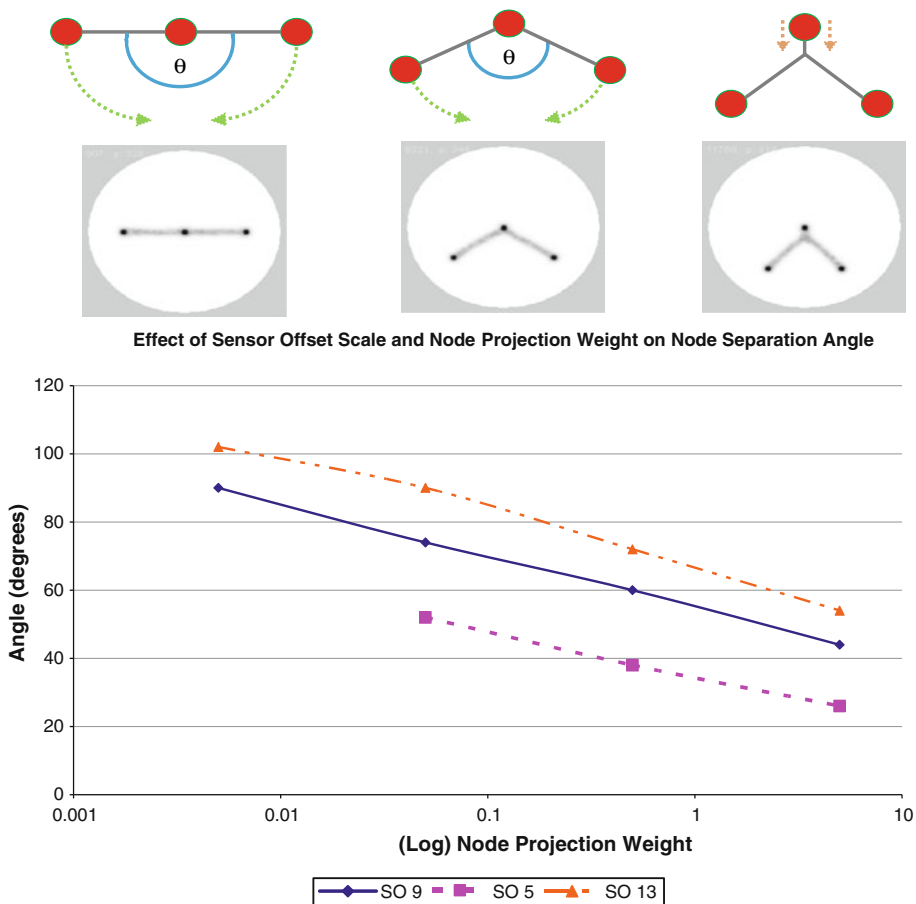


Fig. 11 Measuring the critical angle affecting network evolution between three nodes. *Top row* schematic illustration of experimental evolution. *Middle row* experimental snapshots. Right image shows example just after critical angle has been exceeded and unzipping starts. *Bottom row* plot of critical sensor angle thresholds at different node projection strengths and sensor offset distances

weights could even ‘unzip’ nearby Steiner points, returning the connections to the node itself. The wide region of influence at high node weights is analogous to the influence of different peg diameters in soap film minimisation schemes (Dutta et al. 2008). High node weights result in a stimulus which takes the form of a circular area of relatively large diameter which separates the two outer paths connecting to the circle, thus the critical angle can be reduced to a smaller value before the paths contact and merge. Low node weights result in a smaller stimulus area which is not able to separate the two outer paths to the node, resulting in a larger critical angle before the paths merge.

Although the findings illustrate the critical angle for evolution around a single node of the network it should be emphasised that changes to a single node position can have a significant impact on the evolution of the entire network. This is because when the network configuration changes (by zipping away from a node when a critical angle between two outer nodes is reached) this reconfiguration subsequently affects the node angles at other locations in the network. The effects are compounded when one takes into consideration that the network evolution is continuing at all parts of the network simultaneously. The final configuration (if indeed the work final can be used, since the configuration consists of dynamical network flows) only occurs when the flows in the entire network are balanced.

5.3 Real-time tracking of network topology evolution

Tracking the evolution of transport network connectivity is a difficult process for *Physarum* transport networks and for the emergent transport networks of the particle collective. Both examples are usually represented spatially (photograph of plasmodium network or digital image of transport network). For *Physarum* networks the difficulty is in deciding which information to include or exclude. For example, how to separate the substrate (damp tissue paper or agar, including food and remnants of old/decaying protoplasmic tubes) from the desired network structure. The question of actual network connectivity is also difficult—network paths may meander significantly and differ in thickness. In the case of the model transport networks one needs to automatically detect the network nodes and analyse the network to establish which nodes are connected by network paths. Both *Physarum* and the model networks are both dynamic networks—their patterning and connectivity is continually changing, presenting difficulties in the automatic extraction of network connectivity.

The analysis which provides information about node connectivity in the particle model is generated in the following way. Before an experiment is started the start positions of the nodes are automatically detected by standard thresholding methods and stored, giving the number of network nodes and their co-ordinates. This information is stored in a table, the node connectivity table. Each entry in the table stores a relative strength between the two nodes referenced at the at the node table axis positions.

During an experimental run each particle, initialised at a random position, receives a ‘source node’ identifier when it passes the vicinity (specified by *node_radius*, value of 11 pixels) of the first node it encounters. When the same particle encounters a different node the new node is set as the ‘destination node’ ID. The node connectivity table entry indexed by the two positions (Source ID and Destination ID) is incremented by a certain value, *node_inc* (set to 1) as shown in Fig. 12, top row. Finally the Destination ID becomes the new source ID, and the Destination ID value is cleared until a different node is encountered. This process is repeated for each of the thousands of particles in the lattice and, over time, a ‘picture’ of node connectivity emerges as strength weights in the node connectivity table. The degree of connectivity of a node can be assessed by checking the

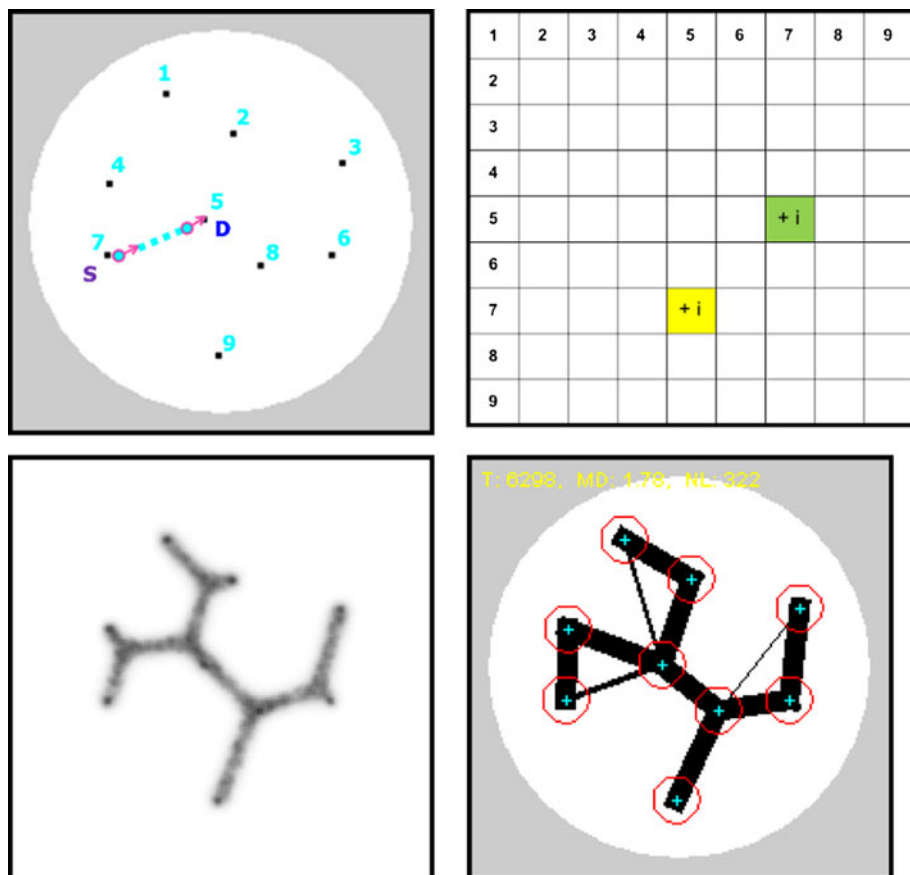


Fig. 12 Illustration of the real-time analysis of emergent transport networks. *Top left* Node positions (1–9) are stored by automatic detection before experiment begins. Particle source node (s) and destination node (d) are indicated. *Top right* Node connectivity table is updated whenever a particle arrives at a different node. *Bottom left* actual configuration of emergent transport network. *Bottom right* Topological connectivity of emergent transport network. *Crosses* indicate node positions and *circles* indicate radius of the particle node detection vicinity. *Thicker lines* between indicate stronger connections. See text for explanation

connectivity strengths along its row in the node table: any nodes greater than *node_threshold* (set to zero), indicates a connection between the nodes. To reduce the memory time of the connectivity of the node table, all the nodes in the table are decremented periodically by a value *node_decay* (with a value of 0.05). Paths which are no longer reinforced will result in falling values at their respective node table entries. This ensures that the information node table can keep up to date with the changing network topology. The result of the network analysis is a transformation of the spatially represented transport network flux into a real-time topological representation of network connectivity (Fig. 12 bottom row). The updating of particle positions it iterated for each particle at every scheduler step and the node table memory decay is executed every 100 steps (this interval is also used to sample the connectivity of the node table for statistics).

Although the actual network and topology are similar, the topological representation shows extra connections (shown as the thin lines between nodes on Fig. 12, bottom right).

This is because at low node projection weights (where node strength is similar to path strength) the transport networks are pulled away from the nodes to adopt the minimal network shape (the Steiner tree, which has additional nodes). In doing so, some of the particles in the network path between three nodes do not actually visit the fixed middle node, instead passing through the Steiner point between the nodes. The topological representation illustrates that a fixed node may be indirectly connected, via a Steiner node, to a nearby node. The relative differences in strength between node connections can be visualised by the differences in thickness of the dynamical representation of network topology.

To explore the dynamic connectivity of the emergent transport networks during the formation and evolution of the networks a small population (p 3, 1,200 particles) was initialised using the filamentous condensation (FC) method at both very low (weight 0.005) and very high (weight 5) node projection weights. Twenty runs at each weight, for each dataset D1, D2 and D3, were carried out and each run lasted for 10,000 scheduler steps. Mean degree of connectivity (the mean number of nodes to which a single node was connected) was measured every 100 scheduler steps and the results are summarised in Fig. 13.

During each experiment the population initially rapidly grew in size and formed a transport network around the nodes. The attraction to the nodes then minimised the transport network configurations. Under high node weight conditions, network connections were strongly attracted to the nodes and the tension pulled the networks to configurations with low mean degree values. Networks with high tension tended to have more cycles than networks with low node weights (low tension). Representative networks from the Toussaint hierarchy are placed to the right of each chart at the vertical positions corresponding to their mean degree. Further to the right are all example subtypes of networks found at the end of each experiment from each dataset at the two node weight extremes (solid boxes are high weight and dashed boxes low weight). It can be seen that the high node weight subtypes appear to match the proximity graph configurations in the Toussaint hierarchy at correspondingly similar mean degree values.

The results from low node weight conditions differed significantly from their respective examples in the Toussaint hierarchy. Despite having fewer cycles, the networks formed under low node weight conditions had a much higher mean degree of connectivity. This is because the relatively low attraction to the nodes results in pulling of the network from the nodes and the formation of Steiner points. The resulting graphs have relatively few cycles and, in most cases, approximate spanning tree structures, but have mean degree values similar to proximity graphs in the Relative Neighbourhood Graph and Gabriel Graph range. This suggests that the addition of Steiner points to the transport networks has the advantage of increasing network connectivity whilst simultaneously minimising the number of network cycles and total network length.

Although the low node weight networks result in more tree-like networks there are a number of examples (particularly in the case of the D3 dataset) where the minimisation of network cycles cannot be completed to give the tree structure (Fig. 14).

The reason that such cyclic networks cannot be minimised is that the arrangement of the network around the nodes results in an arrangement of interior angles which are less than the critical angle at which unzipping can fully occur. One method to overcome the limitation on minimisation is to alternate between high tension nodes (giving cyclic networks) and low tension nodes (reducing the number of cycles). An example of alternating global tension is shown in Fig. 15 where the initial phase was high tension and each alternating phase lasted for 5,000 scheduler steps. The chart line shows fluctuations in mean degree during each alternating phase and characteristic network subtypes at the end of each phase

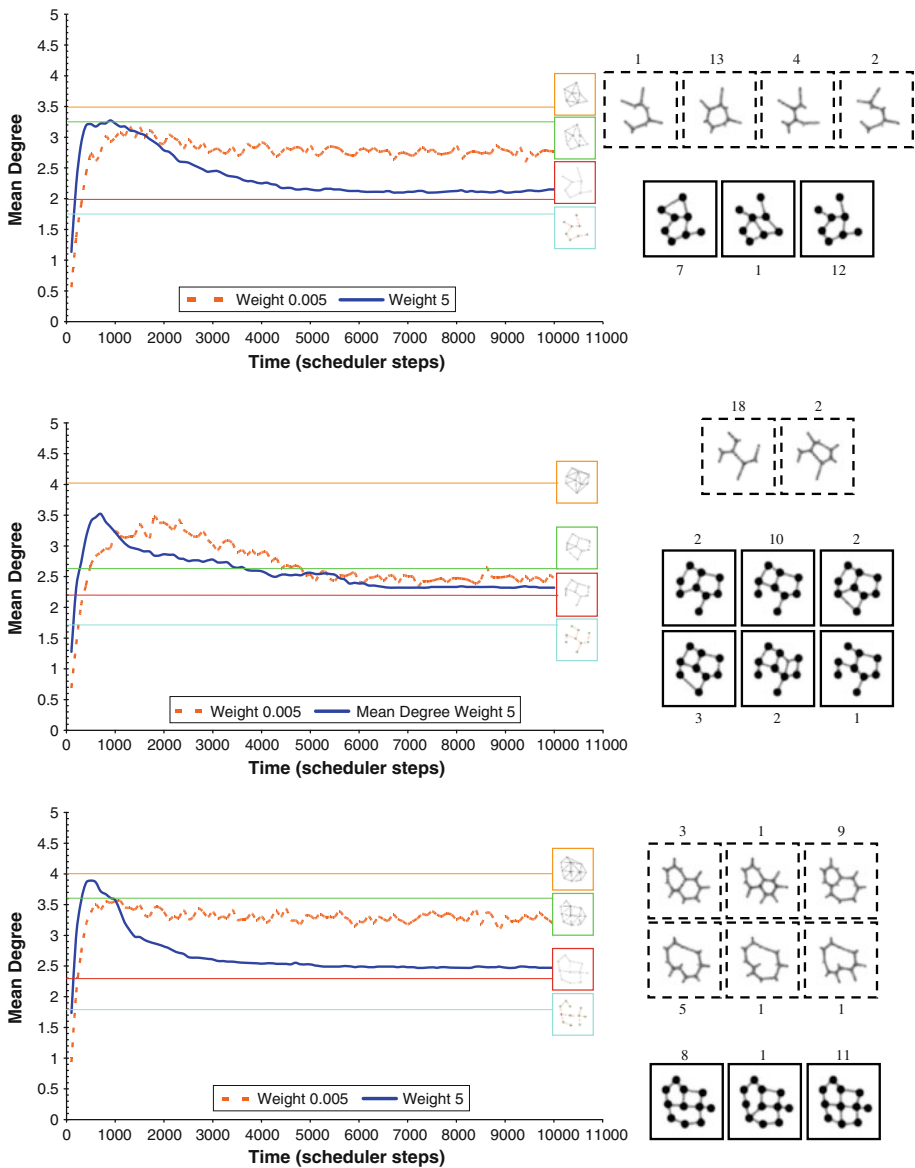


Fig. 13 Evolution of mean degree of network connectivity for datasets D1 (top), D2 (middle) and D3 (bottom) over 10,000 steps, mean from 20 experimental runs. *Left* evolution of mean degree at high (solid, weight = 5) and low (dashed, weight = 0.005) node projection weights. Positions of related proximity graphs indicated at the right side of chart. *Right* All possible final network evolution subtypes observed at low node projection weights (top images, small nodes) and high weights (bottom images, large nodes). Number shows how many examples of subtypes were observed in 20 runs

(the second low tension phase shows two such images as two cycles were closed during this phase). By repeatedly alternating network tension (via node strength) the number of cycles in the network is reduced until a tree structure finally emerges. Why does alternating

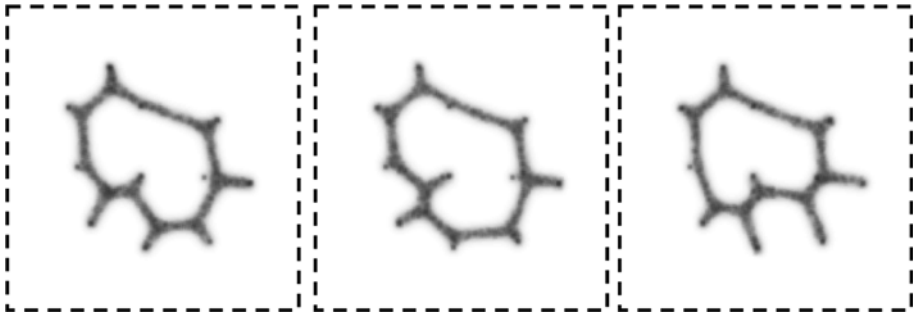


Fig. 14 Examples where minimisation cannot be completed because of large interior angles

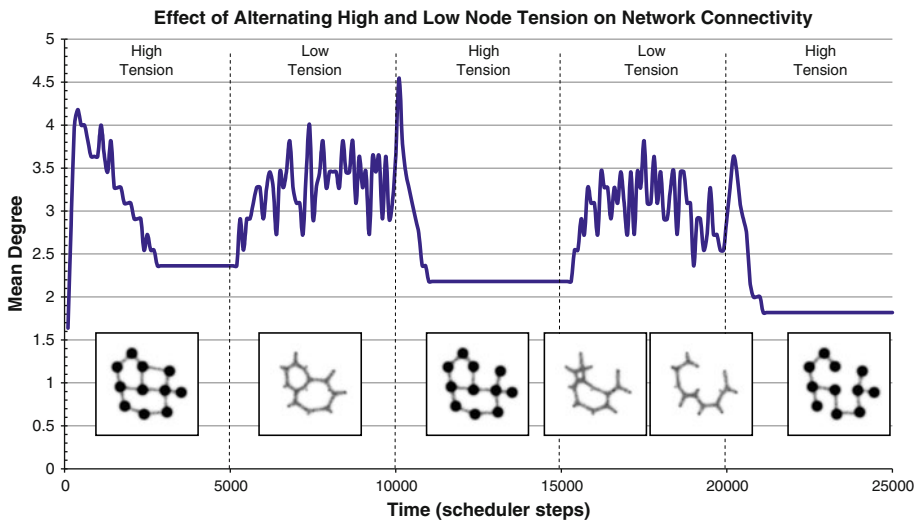


Fig. 15 Alternating high and low tension phases to close network cycles. Main plot line shows mean degree of connectivity. Vertical bars indicate phases in tension (via nutrient concentration) at every 5000 steps. Images show exemplar configurations during each stage as cycles are removed from the network

network tension result in the complete closure of network cycles? We believe this occurs partly because of the initial high tension phase which results in a larger number of smaller network cycles (as opposed to one large cycle). The second reason is that the repeated application of high tension phases temporarily reduces the angles between the nodes, allowing minimisation to occur in the subsequent low tension phase. The chart plot indicates the stepwise nature of the minimisation process showing that at each high tension phase the number of cycles is reduced. The pre-selection of nutrient concentration weights effectively tunes the evolution of the transport network to those with pre-defined characteristics (for example trees or cyclic networks). We are currently exploring the use of the dynamic topology information to apply *local* influences (variations in individual node weights) by a feedback process to guide the evolution into specific configurations for computational geometry tasks.

6 Computational resources and performance

Multi agent based models in general require large computational resources. This is often due to the inherent costs in iterating through large populations of independent entities, the associated costs of switching execution between agents, and communication between agents and their environment. In terms of the *Physarum* model presented herein this cost is also compounded by the fact that the shared agent environment is subject to diffusion of the network trails at each scheduler step. Thus, the computational costs increase significantly not necessarily due to the increased number of points, but due to the increased area size used. Furthermore we are not simply using an agent model to perform the computation, we are utilising the collective emergent properties of the agent interactions (surface tension effects, and minimisation effects) to actually execute what would normally be directly implemented in a software algorithm. In effect, there are two separate orders of computational cost to be considered: The first is the cost of generating the emergent properties of the virtual substrate formed by the agent interactions and environment interactions. It is anticipated that this cost could be reduced by techniques as GPU acceleration by parallelisation of the environmental diffusion and possibly also the agent interactions, since the agent particles could be classed as SIMD operations on the shared data structure (Lysenko and D'Souza 2008; Richmond et al. 2009). The second order cost is in using these emergent properties to approximate the geometry problems, for example the time taken to minimise the surface using the plasmodial shrinkage method. It must be acknowledged that the two separate orders of computation required (simulating the substrate and then using the substrate) do indeed restrict the performance of the model for the targeted problems.

The resources required for the different geometry problems are very similar since the main computational cost is based upon the number of agent particles and the area size of the environment. The main difference between the actual implementation of the 'algorithms' is the different methods of initialising the particle population. As with *Physarum* itself, the model only approximates the solution to the geometry problems and, unlike classical algorithms, cannot be guaranteed to find the optimal solutions to the problems. However, the behaviour of the model does have some interesting and potentially advantageous properties (such as not having to restart the algorithm as new data points are added, or as points are removed) which could prove useful in problems with dynamically changing datasets. If a physical material with properties similar to that of *Physarum* could be utilised this would remove the computational costs associated with simulation of the substrate.

7 Conclusions and scope for further work

We have presented a particle model whose interactions produce complex emergent behaviour resulting in a synthetic virtual material which approximates the network formation and evolution of the plasmodium of *Physarum polycephalum*. Different mechanisms of initialising, manipulating and controlling the pattern formation behaviour of the substance were discussed. The virtual plasmodium was used to construct and evolve complex transport networks and to perform spatially represented approximations of computational geometry problems. We analysed the evolution of the emergent transport networks around food sources and found that food position (and angular relationship between food sources) were critical in guiding network evolution. The evolution of the

networks was also strongly influenced by globally applied differences in the strength of the food sources, as is the case with *Physarum* path choice (Nakagaki et al. 2007) and possibly even as a basis of a search strategy (Latty and Beekman 2009). It is known through the work of Nakagaki et al. (2004a) that *Physarum* generates and evolves networks which exhibit properties of short network length and tolerance to random disconnection. We also found evidence to support previous research by Adamatzky (2008a) which stated that *Physarum* networks traversed the Toussaint hierarchy of proximity graphs during its foraging and minimising behaviour. Since *Physarum* is not aware that it traverses the family of proximity graphs, what could be the reason for the differing network behaviour during the foraging and minimising stages? One possible explanation is that the difference between foraging and minimising behaviour reflects on the relationship between area and distance. In the foraging stage area coverage must be maximised to find suitable food sources (whose locations are, of course, unknown to the plasmodium). In the minimisation stage, however, increased area represents more metabolic work to transport nutrients and so the minimisation of network distance becomes most important. Presumably it is during the minimisation stage when nutrient size and concentration affects the minimisation behaviour and network configuration. The Plasmodial Shrinkage method presents a particularly interesting interface between the fuzzy boundary of area and distance. In reality this hypothetical switch between two modes of operation is under distributed control in the real plasmodium, rather than the simplistic choice seen in the model, and different areas of the plasmodium may even behave in different modes simultaneously. Work is currently underway to utilise the adaptive population size behaviour described in Sect. 5 so that foraging and minimisation behaviour emerge from low-level interactions rather than a coarsely applied external behavioural mode.

Both model and the real organism have flux as the basis for the emergence of complex behaviour. They differ in that the model does not explicitly represent the tube structures through which the protoplasmic flux occurs. There is also no explicit definition of the molecular entities to which the particles might relate. This is partially due to the simplifying nature of the model and partly due to the fact that the internal mechanisms by which *Physarum* generates its emergent behaviour are not well understood, even though the component parts are. Given such unknown factors we adopt a synthetic approach where an attempt is made to synthesise similar emergent behaviour from the bottom-up without constraining rules. Although this only proves the sufficiency of the model to generate the similar behaviour (and not necessarily the actual cause) the model adheres to the low-level notions of very simple components and local sensory and motor interactions which also constrain the real organism. The model and the real organism are also different in that the network configurations in the model are not encumbered by the debris (for example fragments of old protoplasmic tube) which affects *Physarum* (although this can be minimised, as in the case of growing the organism on a water substrate, Adamatzky and Jones 2008). The model does, however, share its most important feature with the real organism: the complex behaviour (network formation, evolution and minimisation) are collective, distributed and emergent properties which cannot be reduced to the specified particle algorithm. Although space constraints restrict detailed discussion, further work is in development which encompasses other abilities such as inertial movement, spontaneous distributed oscillations, amoebic movement and indirect external control of the virtual plasmodium behaviour. We aim to couple these features with biologically plausible growth behaviour to provide a model which may provide some insights into the behaviour of *Physarum* and provide possibilities for *Physarum* inspired computing and robotic devices.

Acknowledgements The author is grateful to Andrew Adamatzky for providing the source data node positions for the proximity graph experiments. The work was supported by the Leverhulme Trust research grant F/00577/1 “Mould intelligence: biological amorphous robots”.

References

- Adamatzky A (1991) Neural algorithm for constructing minimum spanning tree of a finite planar set. *Neural Netw World* 6:335–339
- Adamatzky A (2007) *Physarum* machine: implementation of a Kolmogorov-Uspensky machine on a biological substrate. *Parallel Process Lett* 17(4):455–467
- Adamatzky A (2008a) Developing proximity graphs by *Physarum Polycephalum*: does the plasmodium follow Toussaint hierarchy? *Parallel Process Lett* 19(1):105–127
- Adamatzky A (2008b) Growing spanning trees in plasmodium machines. *Kybernetes* 37(2):258–264
- Adamatzky A, Jones J (2008) Towards *Physarum* robots: computing and manipulating on water surface. *J Bionic Eng* 5(4):348–357
- Adamatzky A, De Lacy Costello B et al (2008) Universal computation with limited resources: Belousov-Zhabotinsky and *Physarum* computers. *Int J Bifurc Chaos* 18(8):2373–2389
- Aono M, Hara M (2007) Amoeba-based nonequilibrium neurocomputer utilizing fluctuations and instability. *Lecture notes in computer science*, vol 4618. Springer, Berlin, pp 41–54
- Duckham M, Kulik L et al (2008) Efficient generation of simple polygons for characterizing the shape of a set of points in the plane. *Pattern Recognit* 41(10):3224–3236
- Dutta P, Khastgir SP et al (2008) Steiner trees and spanning trees in six-pin soap films. *Arxiv*: <http://arxiv.org/abs/0806.1340>
- Galton A, Duckham M (2006) What is the region occupied by a set of points? *Lecture notes in computer science*, vol 4197. Springer, Berlin, pp 81–98
- Gunji YP, Shirakawa T et al (2008) Minimal model of a cell connecting amoebic motion and adaptive transport networks. *J Theor Biol* 253(4):659–667
- Hales T (2001) The honeycomb conjecture. *Discrete Comput Geom* 25(1):1–22
- Hickey D, Noriega L (2008) Relationship between structure and information processing in *Physarum polycephalum*. *Int J Model Identif Control* 4(4):348–356
- Ishiguro A, Shimizu M et al (2006) A modular robot that exhibits amoebic locomotion. *Robot Auton Syst* 54:641–650
- Jaromczyk JW, Toussaint GT (1992) Relative neighborhood graphs and their relatives. *Proc IEEE* 80(9):1502–1517
- Jones J (2008) An emergent pattern formation approach to dynamic spatial problems via quantitative front propagation and particle chemotaxis. *Int J Unconv Comput* 4(4):1–34
- Jones J (2009) Passive vs active approaches in particle approximations of reaction-diffusion computing. *Int J Nanotechnol Mol Comput* 1(3):37–63
- Jones J (2010a) Characteristics of pattern formation and evolution in approximations of *physarum* transport networks. *Artif Life* 16(2):127–153
- Jones J (2010b) The emergence and dynamical evolution of complex transport networks from simple low-level behaviours. *Int J Unconv Comput* 6(2):125–144
- Kobayashi R, Tero A et al (2006) Mathematical model for rhythmic protoplasmic movement in the true slime mold. *J Math Biol* 53(2):273–286
- Latty T, Beekman M (2009) Food quality affects search strategy in the acellular slime mould, *Physarum polycephalum*. *Behav Ecol* 20(6):1160–1167. doi:[10.1093/beheco/arp111](https://doi.org/10.1093/beheco/arp111)
- Lobovkina T, Dommersnes PG et al (2008) Shape optimization in lipid nanotube networks. *The European Phys J E-Soft Matter* 26(3):295–300
- Lysenko M, D’Souza RM (2008) A framework for mega scale agent based model simulations on graphics processing units. *J Artif Soc Soc Simul* 11(4):10
- Nakagaki T, Yamada H et al (1999) Reaction–diffusion–advection model for pattern formation of rhythmic contraction in a giant amoeboid cell of the *Physarum* plasmodium. *J Theor Biol* 197(4):497–506
- Nakagaki T, Yamada H et al (2000) Maze-solving by an amoeboid organism. *Nature* 407:470
- Nakagaki T, Kobayashi R et al (2004a) Obtaining multiple separate food sources: behavioural intelligence in the *Physarum* plasmodium. *Proc R Soc B Biol Sci* 271(1554):2305–2310
- Nakagaki T, Yamada H et al (2004b) Smart network solutions in an amoeboid organism. *Biophys Chem* 107(1):1–5

- Nakagaki T, Saigusa T, Tero A, Kobayashi R (2007) Effects of amount of food on path selection in the transport network of an amoeboid organism. In: Proceedings of the international symposium: topological aspects of critical systems and networks, 13–14 February 2006. World Scientific Publishing, Sapporo, Japan, pp 94–100
- Richmond P, Coakley S, Romano DM (2009) A high performance agent based modelling framework on graphics card hardware with CUDA. In: Proceedings of the 8th international conference on autonomous agents and multiagent systems-volume 2. International Foundation for Autonomous Agents and Multiagent Systems, Budapest, Hungary, pp 1125–1126
- Shirakawa T, Adamatzky A, Gunji YP, Miyake Y (2009) On simultaneous construction of Voronoi diagram and Delaunay triangulation by *Physarum polycephalum*. Int J Bifurc Chaos 19(9):3109–3117
- Takagi S, Ueda T (2007) Emergence and transitions of dynamic patterns of thickness oscillation of the plasmodium of the true slime mold *Physarum polycephalum*. Phys D Nonlinear Phenom 237(3): 420–427
- Takamatsu A (2006) Spontaneous switching among multiple spatio-temporal patterns in three-oscillator systems constructed with oscillatory cells of true slime mold. Phys D Nonlinear Phenom 223(2): 180–188
- Takamatsu A, Takaba E et al (2009) Environment-dependent morphology in plasmodium of true slime mold *Physarum polycephalum* and a network growth model. J Theor Biol 256(1):29–44
- Tero A, Kobayashi R et al (2007) A mathematical model for adaptive transport network in path finding by true slime mold. J Theor Biol 244(4):553–564
- Toussaint GT (1980) The relative neighbourhood graph of a finite planar set. Pattern Recognit 12(4): 261–268
- Tsuda S, Aono M et al (2004) Robust and emergent *Physarum* logical-computing. BioSystems 73(1):45–55
- Tsuda S, Zauner K et al (2007) Robot control with biological cells. BioSystems 87(2–3):215–223