Chapter 21
On Hybrid Classical and Unconventional Computing for Guiding Collective Movement

Jeff Jones

Abstract Collective movement in living systems typically displays complex dynamics which cannot be described by the component parts themselves. Plasmodium of slime mould *Physarum polycephalum* exhibits complex amoeboid movement during its foraging and hazard avoidance which may be influenced by the local placement of attractants, repellents and light irradiation stimuli. Slime mould is a useful inspiration to soft-robotics due to its simple component parts and the distributed nature of its control and locomotion mechanisms. However, it is challenging to interface classical computing devices to a distributed system which utilises self-organised and emergent properties. In this chapter we investigate potential *hybrid* approaches to the task of automatically guiding collective robotics devices, using a multi-agent model of slime mould. We demonstrate a variety of simple open-loop guidance methods. We then describe a hybrid classical/unconventional computing approach using a closed-loop feedback mechanism with attractant and repellent stimuli. Both stimulus types were capable of successful automatic guidance, but we found that repellent stimuli (a light illumination mask) provided faster and more accurate guidance than attractant sources, which were found to exhibit overshooting phenomena at path turns. The method allows traversal of convoluted arenas with challenging obstacles such as narrow channels and complex gratings, and provides an insight into how unconventional computing substrates may be hybridised with classical computing methods to take advantage of the mutual benefits of both approaches.

21.1 Introduction: Collective Movement

Collective movement is a directed movement of multiple individuals which are coupled (directly or indirectly) by some aspect of their environment or special senses. The phenomenon is observed in natural systems which span huge variations in spatial

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

© Springer International Publishing Switzerland 2017

A. Adamatzky (ed.), *Advances in Unconventional Computing*,
Emergence, Complexity and Computation 23,
DOI 10.1007/978-3-319-33921-4_21

scale, temporal scale, and in their environmental medium. At very small spatial scales this includes self-organised movement or aggregation of collectives composed of individual cells to form regular patterns. In bacteria this can be observed, for example, in the formation of regular patterns in *Bacillus subtilis* [36] and *Proteus mirabilis* [43]. Cells of the cellular slime mould *Dictyostelium discoideum* are well known to aggregate under the influence of $_cAMP$ [16, 34]. At slightly larger scales human cells are known to move collectively during embryogenesis [46], wound repair [9] and tumorigenesis [13] in response to a wide range of chemotactic, bioelectric and mechanical stimuli [17, 35].

Collective movement at the population level can result in dynamic and dramatic patterning phenomena, such as swarming [10], flocking [44], herding [18], and shoaling and schooling [22, 42]. The specific biological and generalised coupling mechanisms which generate these emergent patterns from the low-level individual interactions have been studied [15, 51, 61]. In human environments collective movement is seen in walking trail patterns [21], crowd dynamics [20, 67], and car traffic systems [19, 37].

Non-living systems may also exhibit collective movement, for example the phenomenon of sorted patterned ground [31] or the evolution of dune structures [33, 62]. But in these non-living cases the movement of the 'individuals' is passive and undirected, guided only by environmental forces (freeze-thaw cycles and wind transport respectively). In living systems collective movement not only responds to environmental forces but also adds an element of responding to an external stimulus. This directed response enables the dynamical cohesion of a population in space (for example flocking in response to predatory threats), the efficient ordering or movement of a mobile population (trails or raiding fronts of ant colonies), the aggregation of a population towards a single location (for example prior to the assembly of the grex structure in *Dictyostelium*), and the arrangement of complex patterns (the concentric patterns in *B. subtilis* or the arrangement of endothelial precursor cells to form vascular structures). The type of stimulus and the location at which it is presented are an important consideration in the response of the mobile collective.

Collective movement is of interest for computing and robotics applications because the mechanisms which enable collective movement in natural systems are relatively simple, use local communication cues, exploit self-organised patterning and exhibit distributed control. There is typically no central orchestrator of movement in such systems, and the contribution of all entities may be of equal importance. These emergent behaviours result in efficient collectives which contain redundant parts and are resilient to damage or interruption. Simple identical components would reduce the cost of robotic devices. Furthermore, communication between robotic entities and evaluation of current position and future goal position imply a significant computational cost which would be multiplied when the collective contains a large robotic population. The exploitation of strategies exploited by living collective systems in artificial robotic collectives may result in useful physical and computational cost savings, with the benefit of innate autonomy and resilience of the collective. As previously noted, collective movement is observed in a diverse number

of target organisms operating at very different scales. If we are to take inspiration from living examples of collective movement for robotics purposes which example system should be chosen?

21.2 Collective Movement in Slime Mould *Physarum Polycephalum*

An ideal candidate for a biological organism to explore collective movement would be capable of the complex sensory integration, movement and adaptation, yet be composed of a relatively simple component parts that are amenable to simple understanding and external influence. The acellular slime mould *Physarum polycephalum* meets these criteria. *P. polycephalum* is a giant single-celled amoeboid organism, visible to the naked eye. During the plasmodium stage of its complex life-cycle [50] it takes the form of a constantly adapting protoplasmic network. This network is comprised of a sponge-like material which exhibits self-organised oscillatory and contractile activity which is harnessed in the transport and distribution of nutrients within this internal transport network. The organism is remarkable in that the control of the oscillatory behaviour is distributed throughout the almost homogeneous medium and is highly redundant, having no critical or unique components.

The plasmodium is amorphous in shape and ranges from the microscopic scale to up to many square metres in size. It is a single cell syncytium formed by repeated nuclear division, comprised of a sponge-like actomyosin complex co-occurring in two physical phases. The gel phase is a dense matrix subject to spontaneous contraction and relaxation, under the influence of changing concentrations of intracellular chemicals. The protoplasmic sol phase is transported through the plasmodium by the force generated by the oscillatory contractions within the gel matrix. Protoplasmic flux, and thus the behaviour of the organism, is affected by changes in pressure, temperature, space availability, chemoattractant stimuli and illumination [11, 12, 32, 38, 41, 53, 59]. The *P. polycephalum* plasmodium can thus be regarded as a complex functional material capable of both sensory and motor behaviour. Indeed it has been described as a membrane bound reaction-diffusion system in reference to both the complex interactions within the plasmodium and the rich computational potential afforded by its material properties [7]. The study of the computational potential of the *P. polycephalum* plasmodium was initiated by Nakagaki et al. [39] who found that the plasmodium could solve simple maze puzzles. This research has been extended and the plasmodium has demonstrated its performance in, for example, path planning and plane division problems [47, 48], spanning trees and proximity graphs [1, 2], simple memory effects [14, 45], the implementation of logic gates and adding circuits [55, 63].

Robotics use of *P. polycephalum* is possible by exploiting its response to changing conditions within its environment. The migration of the plasmodium is influenced by a wide number of external stimuli including chemoattractants and chemorepellents

[64], light irradiation [40], thermal gradients [65], substrate hardness [54], tactile stimulation [5], geotaxis [66] and magnetotaxis [49]. By careful manipulation of such external stimuli the plasmodium may be considered as a prototype robotic micro-mechanical manipulation system, capable of simple and programmable robotic actions including the manipulation (pushing and pulling) of small scale objects [8, 57], transport and mixing of substances [4]. A *Physarum*-inspired approach to amoeboid robotics was demonstrated by Umedachi et al. [60] in which an external ring of coupled oscillators, each connected to passive and tune-able springs was coupled to a fluid filled inner bladder. The compression of the peripheral springs mimicked the gel contractile phase and the flux of sol within the plasmodium was approximated by the coupled transmission of water pressure to inactive (softer) springs, thus deflecting the peripheral shape of the robot. The resulting movement exhibited flexible behaviour and amoeboid movement.

Physarum has been shown to be a useful model organism in the study of distributed robotics. In this article we explore the problem of collective guidance, i.e. how to move and guide a population of independent mobile entities along a pre-determined path. This task represents a only small subset of general robotics challenges which also include the problem of how to survey and map an unknown environment, and how to plan paths between two or more locations in an environment. Approaches to robotics guidance and planning problems directly inspired by *Physarum* include the simultaneous localisation and mapping problem [30], the generation and manual guidance of collective transport [29], and amoeboid movement [28]. Hybrids of unconventional computing and classical computing substrates are relatively uncommon. In the work of [6] a hybrid path planning system was implemented by using waves from a chemical reaction-diffusion processor to represent start points, end points and obstacles. These waves were used to generate a repulsive field which was used to guide a robot along the arena. The *Physarum* plasmodium itself was used as a guidance mechanism in a biological mechanical hybrid approach where the response of the plasmodium to light irradiation stimuli provided by extended sensors from a classical robot device was then used to provide feedback control to the robot's movement actuators [58]. More recently the problem of generating a path between two points in an arena was tackled with a *Physarum*-inspired morphological adaptation approach [27].

In this article we take the next logical step in these robotics challenges by tackling the problem of dynamically guiding a collective of mobile entities along the path whilst avoiding obstacles. We examine the multi-agent virtual plasmodium and its response to stimuli in Sect. 21.3. Section 21.4 demonstrates simple open-loop examples of guidance. Closed-loop approaches involving a hybrid of classical and unconventional substrates are presented in Sect. 21.5 with assessments of both attractant and repellent guidance methods, novel properties seen during path traversal, and a recovery mechanism for any collectives which may become detached from the target path. We conclude in Sect. 21.6 with a summary of the approach, its main properties and contribution to the field.

21.3 A Virtual Collective Inspired by Slime Mould

To explore guided collective movement we use a simple modification to the particle approximation of *P. polycephalum* introduced in [24] which generated dynamical adaptive transport networks. The approach is based on the concept of simple component parts which exhibit collective emergent behaviour and has shown to be successful in reproducing a wide range of behaviour seen in *P. polycephalum*. Presentation of external environmental stimuli (both attractant and repellent) has been shown to be a critical factor in the evolution of patterning and complexity of computational behaviour within this model [26] (for more information, see [25]). In this approach the plasmodium is represented by a population of mobile particles with very simple behaviours, within a 2D diffusive environment. A discrete 2D lattice stores particle positions and also the concentration of a local generic chemoattractant. The chemoattractant concentration represents the hypothetical flux of sol within the plasmodium. Free particle movement represents the sol phase of the plasmodium and particle positions represent the fixed gel structure (i.e. global pattern) of the plasmodium. Particles act independently and iteration of the particle population is performed randomly to avoid introducing any artifacts from sequential ordering. Particle behaviour is divided into two distinct stages, the sensory stage and the motor stage. In the sensory stage, the particles sample their local environment using three forward biased sensors whose angle from the forward position (the sensor angle parameter, *SA*, set to 90°), and distance (sensor offset, *SO*, set to 15 pixels) may be parametrically adjusted (Fig. 21.1). The offset sensors represent the overlapping filaments within the plasmodium, generating local coupling of sensory inputs and movement to form networks of particles. The *SO* distance is measured in pixels and the coupling effect increases as *SO* increases.

During the sensory stage each particle changes its orientation to rotate (via the parameter rotation angle, *RA*, set to 45°) towards the strongest local source of chemoattractant. After the sensory stage, each particle executes the motor stage

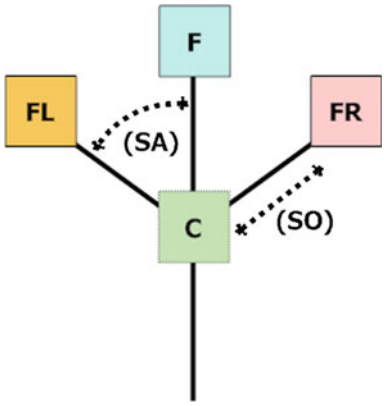

Fig. 21.1 Schematic illustration of a single agent particle. Position on the 2D lattice is indicated by 'C', three offset sensors 'FL', 'F' and 'FR' sample the local chemoattractant concentration, Sensor Offset *SO* parameter sets distance from 'C' to sensors, Sensor Angle *SA* indicates angle between sensors

and attempts to move forwards in its current orientation (an angle from 0–360°) by a single pixel. Each lattice site may only store a single particle and particles deposit chemoattractant (5 arbitrary units) into the lattice only in the event of a successful forwards movement. If the next chosen site is already occupied by another particle the default (non-oscillatory) motor behaviour is to abandon the move, remain in the current position, and select a new random direction.

For experiments with varying population sizes, adaptation of the population size was implemented via tests at regular intervals. The frequency at which the growth and shrinkage of the population was executed determined the turnover rate for the population. The frequency of testing for growth was given by the G_f parameter and the frequency for testing for shrinkage is given by the S_f parameter (both set to 15). Growth of the population was implemented as follows: If there were between G_{min} (0) and G_{max} (10) particles in a local neighbourhood (window size given by G_w, 9 pixels) of a particle, and the particle had moved forward successfully, a new particle was created if there was a space available at a randomly selected empty location in the immediate 3×3 neighbourhood surrounding the particle.

Shrinkage of the population was implemented as follows: If there were between S_{min} (0) and S_{max} (24) particles in a local neighbourhood (window size given by S_w, 5 pixels) of a particle the particle survived, otherwise it was deleted. Deletion of a particle left a vacant space at this location which was filled by nearby particles (due to the emergent cohesion effects), thus causing the blob to shrink slightly.

Diffusion of the collective chemoattractant signal is achieved via a simple 3×3 mean filter kernel with a damping parameter (set to 0.1) to limit the diffusion distance of the chemoattractant. The low level particle interactions result in complex pattern formation. The population spontaneously forms dynamic transport networks showing complex evolution and quasi-physical emergent properties, including closure of network lacunae, apparent surface tension effects and network minimisation. An exploration of the possible patterning parameters was presented in [23].

21.3.1 Generation of Multi-agent Cohesion

Condensation of the multi-agent networks forms uniform sheet-like structures. Figure 21.2 shows the evolution of the stable *SA* 45°, *RA* 45° network within a circular arena. The agents coalesce into network trails and the contraction behaviour condenses the network until all interior space is removed and a sheet-like mass remains. This sheet configuration also exhibits unusual properties: the sheet itself forms a minimal surface shape and ripple-like activity can be seen to propagate through the sheet. The sheet also shows relatively stable dissipative ‘islands’ of greater trail flow. The islands reflect areas where a temporary vacancy of agents exists. The number and size of the islands is related to the sensor offset distance (*SO*) of the agents. When the *SO* parameter increases, the number of vacancy islands decreases and the spacing between them increases (Fig. 21.2). This suggests that the vacancy islands

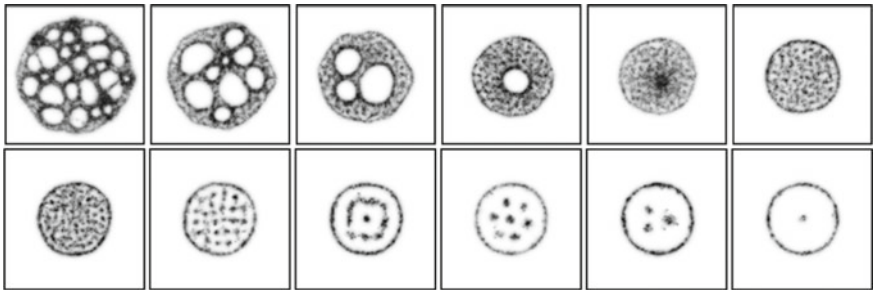

Fig. 21.2 Formation of sheet-like structures and the emergence of dissipative vacancy 'islands'. *Top row* (*left* to *right*) Network evolution over time: $\%p = 20$ agent trails, *SA* 45°, *RA* 45°. *Bottom row* (*left* to *right*) Dissipative vacancy island patterns at *SO*: 9, 13, 19, 23, 28, 38

self-assemble under the influence of the *SO* parameter and represent transient regions of free movement within the mass of particles.

21.3.2 Generation of Oscillatory Dynamics

Although the particle model is able to reproduce many of the network based behaviours seen in the *P. polycephalum* plasmodium such as spontaneous network formation, shuttle streaming and network minimisation, the default motor behaviour does not exhibit oscillatory phenomena and inertial surging movement, as seen in the organism. This is because the default action when a particle is blocked (i.e. when the chosen site is already occupied) is to randomly select a new orientation, resulting in very fluid network evolution. The oscillatory phenomena seen in the plasmodium are thought to be linked to the spontaneous assembly/disassembly of actomyosin and cytoskeletal filament structures within the plasmodium which generate contractile forces on the protoplasm within the plasmodium. The resulting shifts between gel and sol phases prevent (gel phase) and promote (sol phase) cytoplasmic streaming within the plasmodium. To mimic this behaviour in the particle model requires only a simple change to the motor stage. Instead of randomly selecting a new direction if a move forward is blocked, the particle increments separate internal co-ordinates until the nearest cell directly in front of the particle is free. When a cell becomes free, the particle occupies this new cell and deposits chemoattractant into the lattice.

The effect of this behaviour is to remove the fluidity of the default movement of the population. The result is a surging, inertial pattern of movement dependent on population density (the population density specifies the initial amount of free movement within the population). The strength of the momentum effect can be adjusted by a parameter (*pID*) which determines the probability of a particle resetting its internal position coordinates, lower values providing stronger inertial movement. When this simple change in motor behaviour is initiated surging movements are

seen and oscillatory domains of chemoattractant flux spontaneously appear within the virtual plasmodium showing characteristic behaviours: temporary blockages of particles (gel phase) collapse into sudden localised movement (solation) and vice versa. The oscillatory domains themselves undergo complex evolution including competition, phase changes and entrainment. We utilise these dynamics below to investigate the possibility of generating useful patterns of regular oscillations which may be coupled to provide motive force.

21.3.3 Generation of Collective Amoeboid Movement

In the absence of stimuli the oscillatory waves propagating through the particle population (a large cohesive 'blob') simply deform the boundary of the blob. This distortion may cause random movement of the blob through the lattice but this movement is not predictable and is subject to changes in direction. To move the blob in any meaningful way it is necessary to distort the blob with regular stimulus inputs in order to shift its position in a chosen direction. These input stimuli are inspired by stimuli which have been shown to influence the movement of slime mould, attractant stimuli and repellent stimuli. Slime mould is known to migrate towards diffusing attractant stimuli, such as nutrient chemoattractant gradients or increasing thermal gradients. Conversely the organism is known to be repelled (moving away from) certain hazardous chemical stimuli and exposure to light irradiation.

Attractant stimuli are represented in the multi-agent model by the projection of spatial values into the diffusive lattice. Since the particles also deposit and sense values from this lattice they will be attracted towards locations which present the same stimulus. An example of the migration of a self-oscillating blob of multi-agent particles towards attractant stimuli can be seen in Fig. 21.3 in which a blob comprising 4000 particles is exposed to the attractant field generated by projection of four discrete attractant stimuli into the diffusive lattice. The initially random distortion of the blob (Fig. 21.3a) is followed by migration of particles at the leading edge of the blob (i.e. closest to the nearest stimulus, Fig. 21.3b). This changes the shape of the blob and causes travelling waves to emerge, moving forwards in the direction of the nearest nutrient, shifting the position of the blob and moving it towards the nutrient. As each nutrient is engulfed it is 'consumed' simply by decrementing the amount projected into the lattice (Fig. 21.3c, d). Removal of the first stimulus point then exposes the blob to the next point, and so on, causing the blob to migrate along the nutrient locations (Fig. 21.3e–h).

Repellent chemical stimuli may be approximated in the model by projecting negative values into the lattice at spatial sites corresponding to repellents, causing the blob to move away from the stimuli. Alternatively, the light irradiation response may be represented by reducing values within the lattice at exposed regions (the shaded region in Fig. 21.4d) whilst reducing the values sampled by agents' sensors within exposed regions. This reduction of stimuli at exposed regions renders them less attractive to individual particles and particles migrate away from these regions.

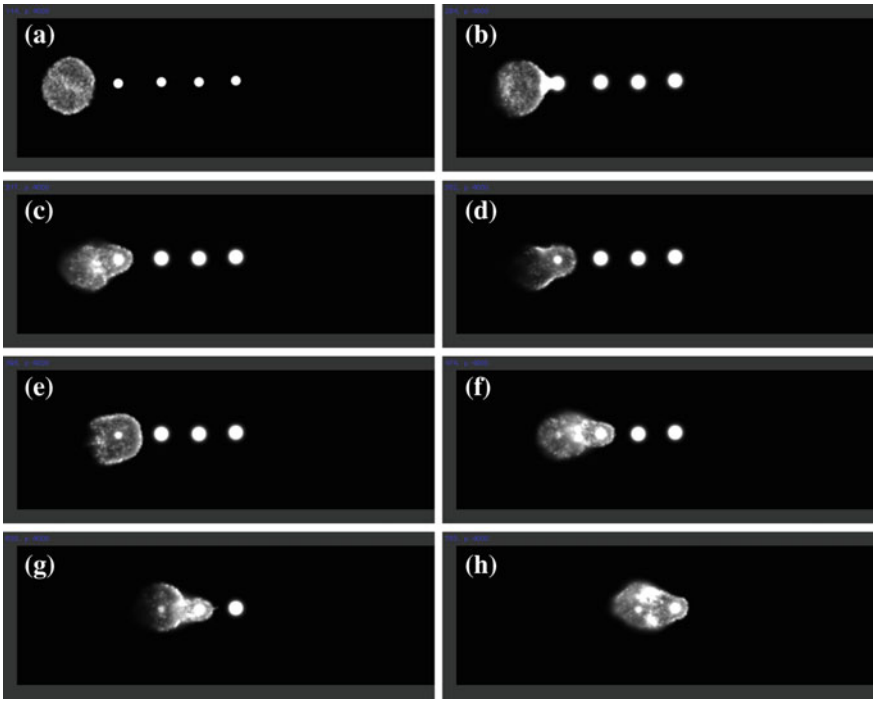

Fig. 21.3 Propagation of self-oscillating blob in the direction of nutrient attractants. **a** Blob comprising of 4000 particles is inoculated on the *left* side of a diffusive lattice containing four nutrient attractants, **b–c** blob shape is deformed as particles move towards the attractants, generating travelling waves within the blob, **d–h** consumption of nutrients exposes the blob to nearby attractants causing the collective to move to the *right* of the arena

This migration is initiated at the interface between exposed and unexposed regions, causing an efflux of particles from exposed regions (Fig. 21.4e). The inherent cohesion of the agent population results in a distortion of the blob shape and a collective movement away from the exposed area 21.4f–h).

In this section we have demonstrated the innate cohesion of the multi-agent population in both oscillatory and non-oscillatory movement modes. The non-oscillatory mode is relatively predictable and attempts to retain a minimal shape profile whilst the oscillatory mode has the advantage of generating self-organised amoeboid movement, at the expense of more unpredictable movement patterns. We have demonstrated how the blob can be influenced by both attractant and repellent stimuli. In the next sections we explore mechanisms to automatically guide the multi-agent population along a pre-selected path.

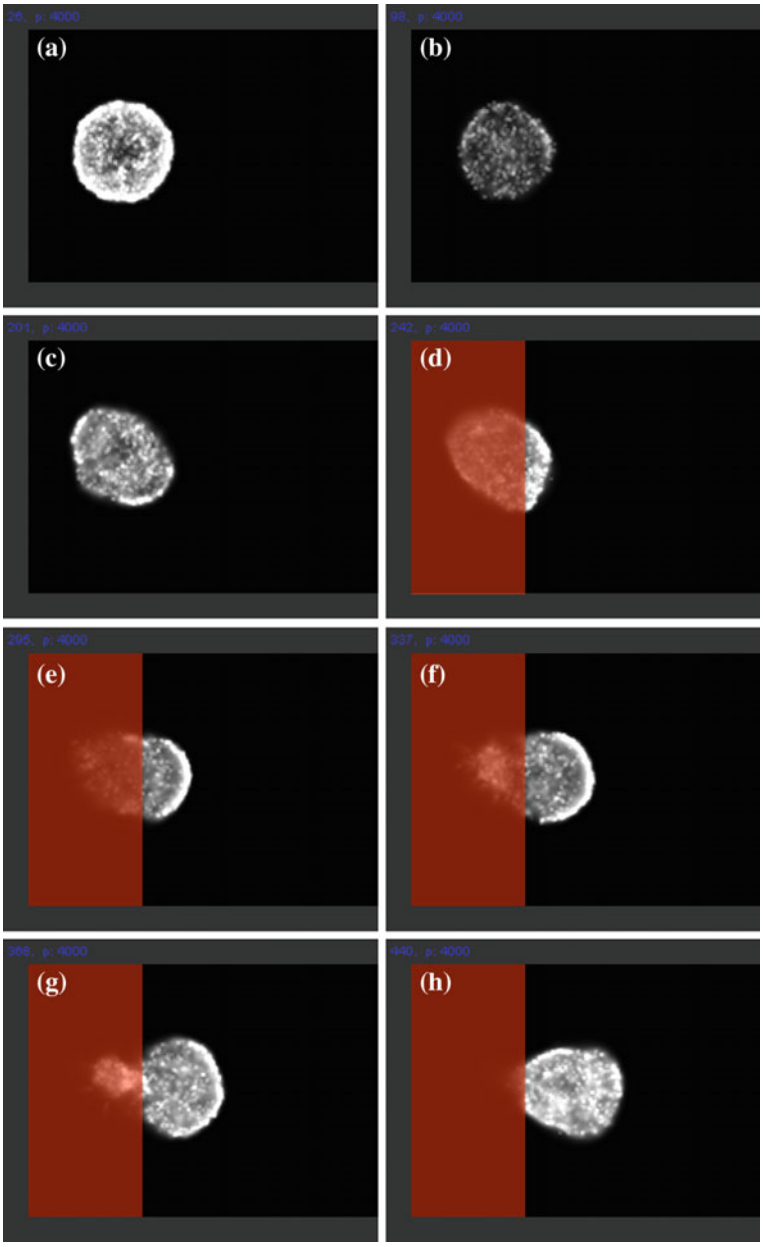

Fig. 21.4 Propagation of self-oscillating blob away from repellent stimulus. **a–c** blob stays in approximately the same position when no stimulus is present, **d** projection of simulated light irradiation to *left* side of the blob (*shaded*) reduces flux and particle sensitivity in the diffusive lattice at these exposed regions, **e–h** particles migrate away from exposed regions towards unexposed regions of the blob, shifting the mass of the blob away from the illuminated region

21.4 Automatically Guided Movement: Open-Loop Methods

In this section we describe relatively simple open-loop approaches to guidance of the virtual plasmodium through an arena. These approaches do not require any input from conventional computing methods and can be implemented completely by projection of spatial patterns to the unconventional computing substrate.

21.4.1 Fuse Method

In this method the entire path is presented as a continuous line of attractant stimuli and a small population is inoculated at the region indicated by the cross-hair (Fig. 21.5a). The population is attracted by the path, growing in size and moving towards it. The stimulus comprising the path is consumed by the population and the population adapts to the consumption of the path by migrating along the path until the end point is reached (Fig. 21.5b–f).

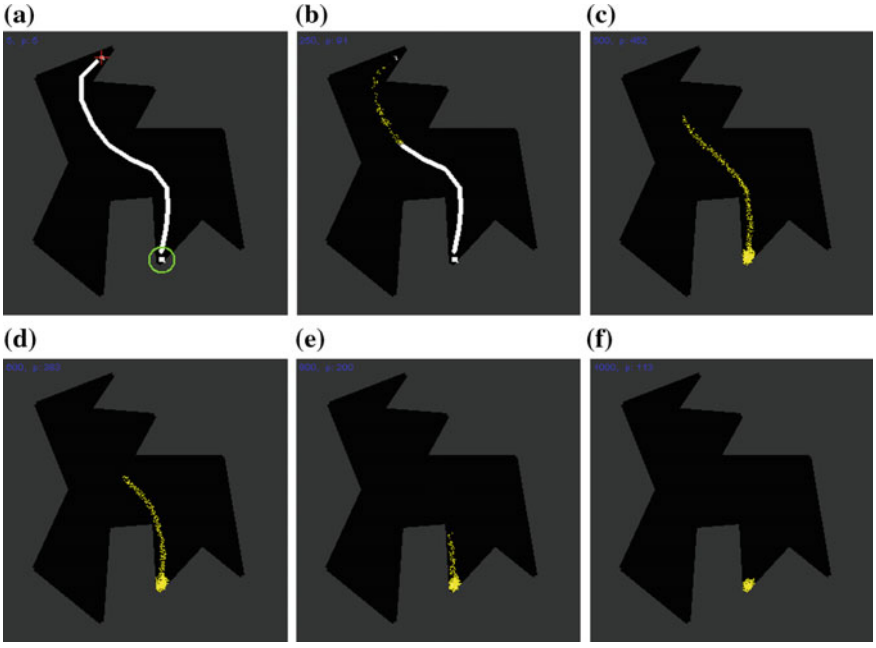

Fig. 21.5 Fuse method. **a** A small population was inoculated at the start point (*cross*) of a path represented by a continuous line of stimuli, **b–c** the population consumes the line, growing and moving along the stimulus path, **d–f** the movement and consumption continues until the blob arrives at the final destination point

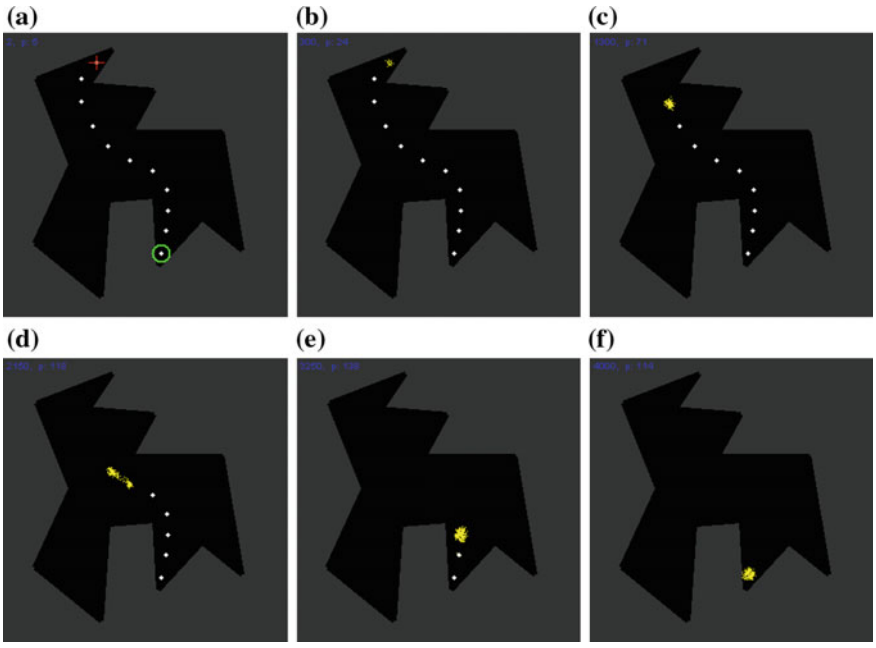

Fig. 21.6 Stepping-stone method. **a** A small population was inoculated at the start point (*cross*) of a discrete array of points, **b–c** the blob consumes each point and is then attracted by the next stimulus point along the path, **d–f** the movement and consumption continues until the blob arrives at the final destination point

21.4.2 Stepping-Stone Method

In the stepping-stone method the path is presented as a sequence of discrete stimulus points (Fig. 21.6a). A small population is again inoculated at the cross-hair location. The small blob of virtual plasmodium consumes the stimulus on the current point on the path and is then attracted to the next point in the path (Fig. 21.6c, d). The blob traverses the path in a series of hops, moving from point to point, until the final goal point is reached (Fig. 21.6f).

21.4.3 Elastic Method

This method utilises the morphological adaptation inherent within the population. The path is again represented by a series of discrete stimuli and a small blob is inoculated at the cross-hair location (Fig. 21.7a). Unlike the fuse and stepping-stone methods, each stimulus point is not immediately consumed, instead the population grows in size to span all points on the path from beginning to end (Fig. 21.7a–c).

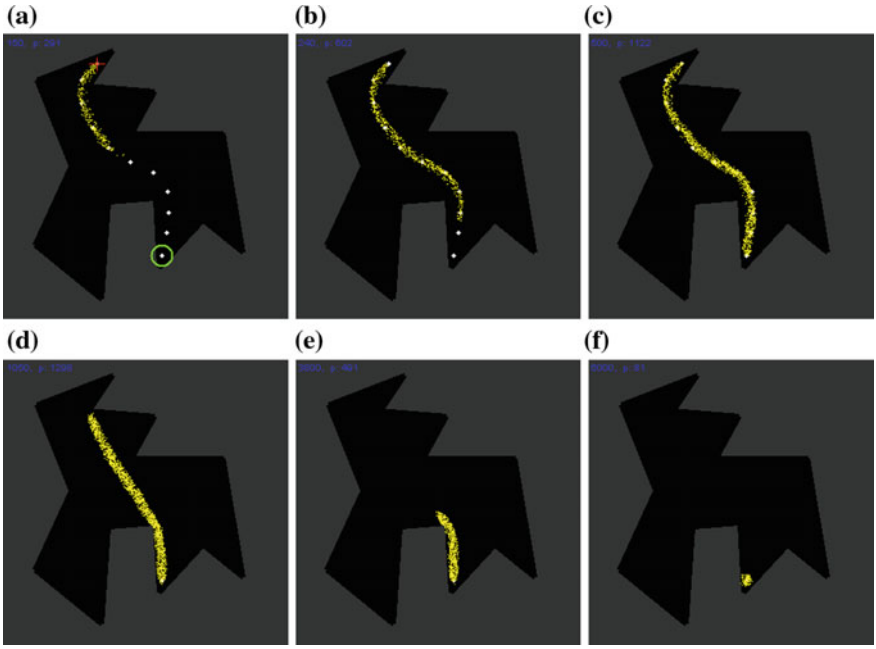

Fig. 21.7 Elastic method. **a–c** A small population was inoculated at the start point (*cross*) and, attracted by the stimulus points of the path, grew to extend across all points, **d–f** all stimulus points except the end point were then removed and the virtual plasmodium adapted to the change of stimulus profile by retracting towards the end point

After reaching the final path point all previous path points are deleted and the blob, which is now only 'anchored' to the attractant of the final point, adapts its shape, shrinking and moving its body plan to the final point, completing the movement.

21.5 Automatically Guided Movement: Closed-Loop Feedback Methods

The open-loop methods, although effective in these examples, do not account for the possibility that the virtual plasmodium may become detached from the path. Once each particular method is set in action, there is no guarantee of the success, or failure, of the motion along the path. We require feedback about the position of the model population within its environment, and mechanisms to dynamically influence the position within the arena to reach the goal. This requires a more complex closed-loop approach.

To provide a real challenge for this approach we utilise the self-oscillatory behaviour of the model plasmodium. This behaviour was introduced in [56] to reproduce

the spontaneous and self-organised oscillation patterns observed within small samples of *Physarum* plasmodia [52]. In the model plasmodium these oscillations emerge from interruptions of individual particle movement, as described in Sects. 21.3.2 and 21.3.3. We adjust the momentum of the self-oscillatory amoeboid movement time using the *pID* parameter. Higher values of *pID* (for example 0.05) result in less persistence of direction of individual particles, whereas lower values (for example 0.001) result in much stronger persistence of direction. The accumulation of interruptions in movement of individual particles results in travelling waves of flux forming within the mass of particles as particles occupy vacant spaces within the collective. It was shown in [28] that these travelling waves could shift the mass of particles, effectively moving the blob of virtual plasmodium. In the same paper it was demonstrated how the self-oscillatory dynamics could be influenced by the manual placement of attractant stimuli and simulated light irradiation stimuli, causing the blobs to move towards attractants and away from light hazards.

21.5.1 Momentum Parameter: Effect on Blob Migration

The effect of initiating self-oscillatory behaviour and its *pID* parameter on blob positions is shown in Fig. 21.8 which details experiments with decreasing *pID* parameters and the effect this has on the random migration of an unstimulated self-oscillating blob from its inoculation position (circle) at the centre of the experimental arena. The position of the blob is recorded as the centroid of all the particles comprising the blob and its path is indicated by the line extending from the central point (Fig. 21.8a, c and e).

For the control condition, with no self-oscillatory behaviour initiated, there was very little displacement of the blob from the inoculation site (maximum of 1.41 pixels displacement over 5000 scheduler steps). Any displacement of the blob was caused by the stochastic influences on individual particles within their sensory method. When oscillatory behaviour was initiated, the build-up of momentary interruptions of particle movement caused the position of the blob to be displaced significantly by the emergent travelling waves within the blob. At *pID* 0.1 the distance from the inoculation site increased gradually (Fig. 21.8b). Decreasing *pID* resulted in greater displacement of the blob (Fig. 21.8d and f). Lower *pID* parameter not only resulted in greater movement from the inoculation site but also more persistence in the direction of movement. This was often followed by sudden random changes in direction (for example, Fig. 21.8e). These changes in direction are the cause of the apparent decrease in error from the original position (Fig. 21.8d and f) at decreased *pID*.

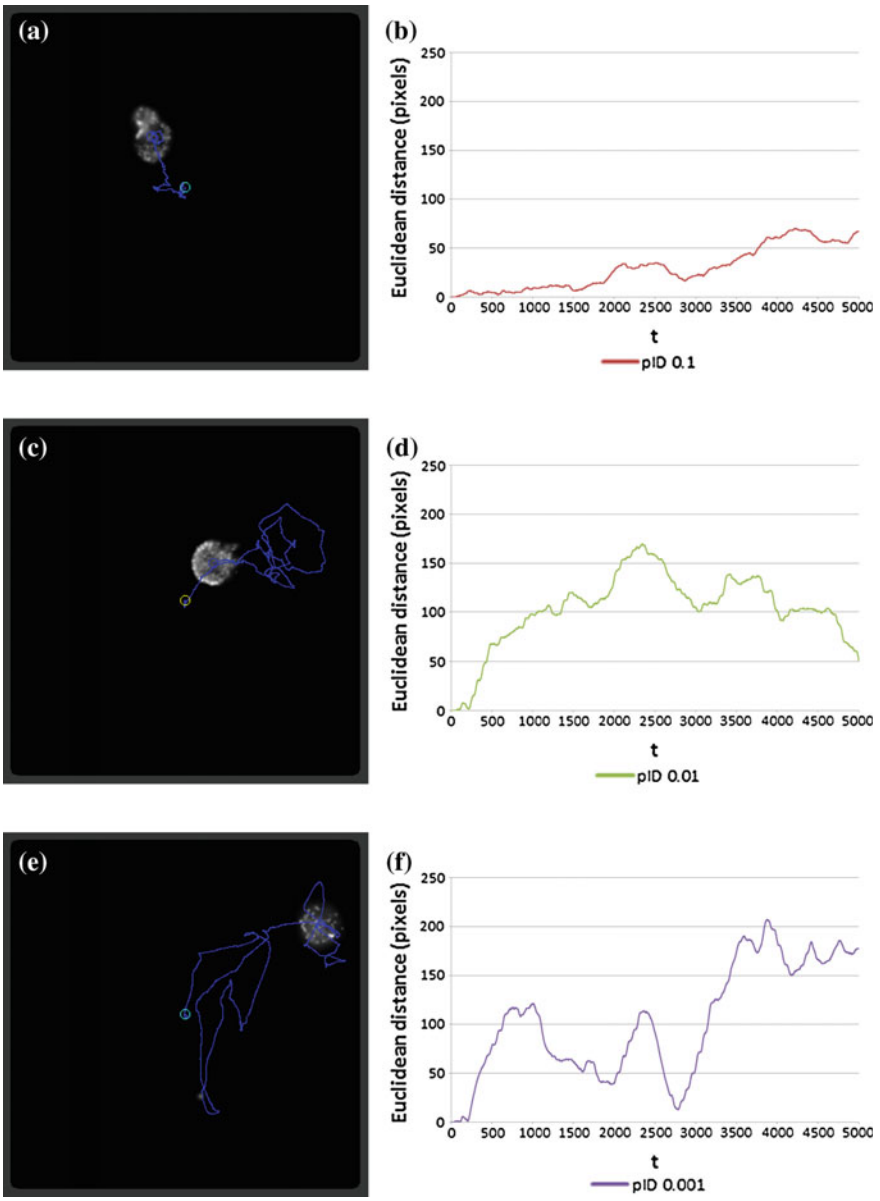

Fig. 21.8 Comparison of spontaneous migration error from inoculation position at different *pID* settings, blob position and error tracked over 5000 steps. **a** record of blob at *pID* 0.1, **b** plot of migration error at *pID* 0.1, **c–d** *pID* 0.01, **e–f** *pID* 0.001

21.5.2 Hybrid Control System

The unpredictability of the movement in the self-oscillating blobs renders it challenging to control and guide their movement. A method for automatic guidance must represent a hybrid approach between the unconventional computing methods which generate the emergent behaviour in the virtual plasmodium (the generation of self-oscillatory travelling waves and amoeboid movement), and classical computing methods to detect the position of the blob and provide the feedback stimuli to guide the blob along the chosen path. A schematic overview of the closed-loop hybrid system is given in Fig. 21.9.

Note that Fig. 21.9 is partitioned by a vertical dashed line. This line indicates the separation of conventional and unconventional approaches and also indicates regions where both approaches interact. The unconventional part of the system generates the emergent oscillatory behaviour of the blob from local and self-organised particle interactions. Information about the blob's collective state is then extracted by the conventional (classical) part of the method which calculates the centroid (centre of mass) of the blob. The position of the blob is compared at every 50 scheduler steps to the points comprising the path in the arena. When the blob is closer to the next point along the path than to the current stimulus location, the *next* point along the path is then selected to provide the new location stimulus for the blob. The target stimulus is then projected into the spatially implemented unconventional part of the method. This stimulus acts to guide the blob towards this new location. Two possible

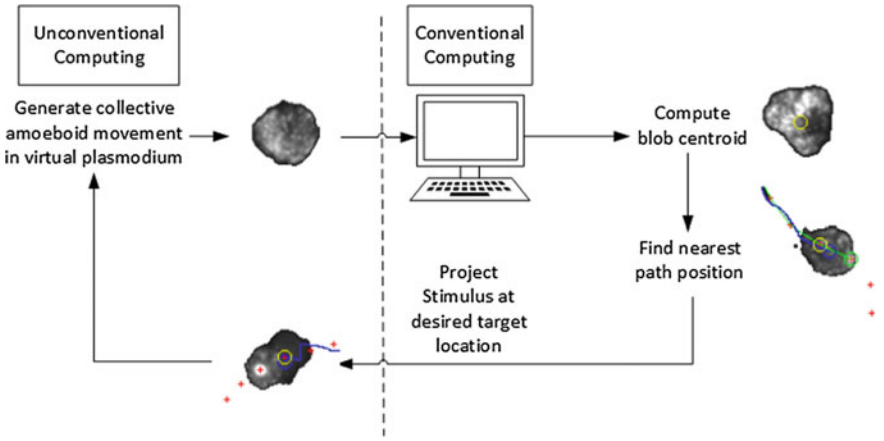

Fig. 21.9 Schematic overview of closed-loop guidance method indicating interface between unconventional computing and conventional computing approaches (*dashed line*). *Left* side of *dashed line* represents the contribution of the unconventional approach, characterised by bottom-up generation of self-organised emergent phenomena, *right* side of *line* represents the contribution of classical computing approaches in generating the control method. Both sides contribute inputs to each other

stimulus types can be used, both of which are seen in the *Physarum* plasmodium and are described in the following sections.

21.5.3 Automatic Guidance with Attractant Stimuli

The first stimulus method used attractant stimuli. In the wild, slime mould is attracted to certain stimuli such as nutrients or localised warm areas. In the model plasmodium attractant stimuli are represented by projecting localised stimuli into the diffusive lattice. These stimuli, when projected within sensory range of the blob, attract the particles comprising the blob. Particles at the outer periphery of the blob move towards the stimulus and the cohesion of the blob (caused by the indirect coupling of the particles' offset sensors) generates travelling waves inside the blob which move directly towards the stimulus. As demonstrated by manual placement in [28], this can be used to guide (or 'pull') the blob *towards* the chosen direction. In the closed-loop method described in this article we can replace the manual placement of stimuli with automated transient placement of point stimuli to guide the blob along the chosen path.

Figure 21.10 shows the results of automated closed-loop guidance of the self-oscillatory blob along a pre-defined path by the attractant method. The path starts at the large cross marker and ends at the circle marker, and individual guidance points on the path are denoted by small crosses (Fig. 21.10a). The path is composed of multiple links between start and end points in an arena populated by solid obstacles (grey shapes) which the agent particles cannot cross. The particle population, comprising 2000 particles, was inoculated at the initial cross marker position and for the first 1000 scheduler steps the blob was allowed to form and stabilise. After 1000 steps the automated guidance mechanism described in the previous sub-section was initiated. The short straight lines connecting the small crosses indicates the pre-defined path (green, online) and the path taken by the oscillatory blob is indicated by the blue (online) markers. The example results include four different momentum (*pID*) parameter settings.

Although the blob follows the path in all examples, at low *pID* (i.e. high momentum) settings there is considerable deviation from the desired path, particularly when the path changes direction (see, for example, Fig. 21.10a and the supplementary video recordings at http://uncomp.uwe.ac.uk/jeff/automatedguidance.htm).

This overshooting of the desired path is caused by the blob position being influenced by the strong oscillatory waves within the blob. At low *pID* values this momentum is particularly strong, causing the blob to overshoot the corners after the momentum of oscillatory waves has been established during straighter sections of the course. Under the strongest momentum condition the blob 'crashes' into the circular obstacle at the lower-right of the arena and the blob has to re-form before its progress can continue. An indication of the strength of the momentum can be seen at the end of each course in Fig. 21.10 where the blob continues to receive attractant input from the final position on the path. Although the position of this stimulus is static, the

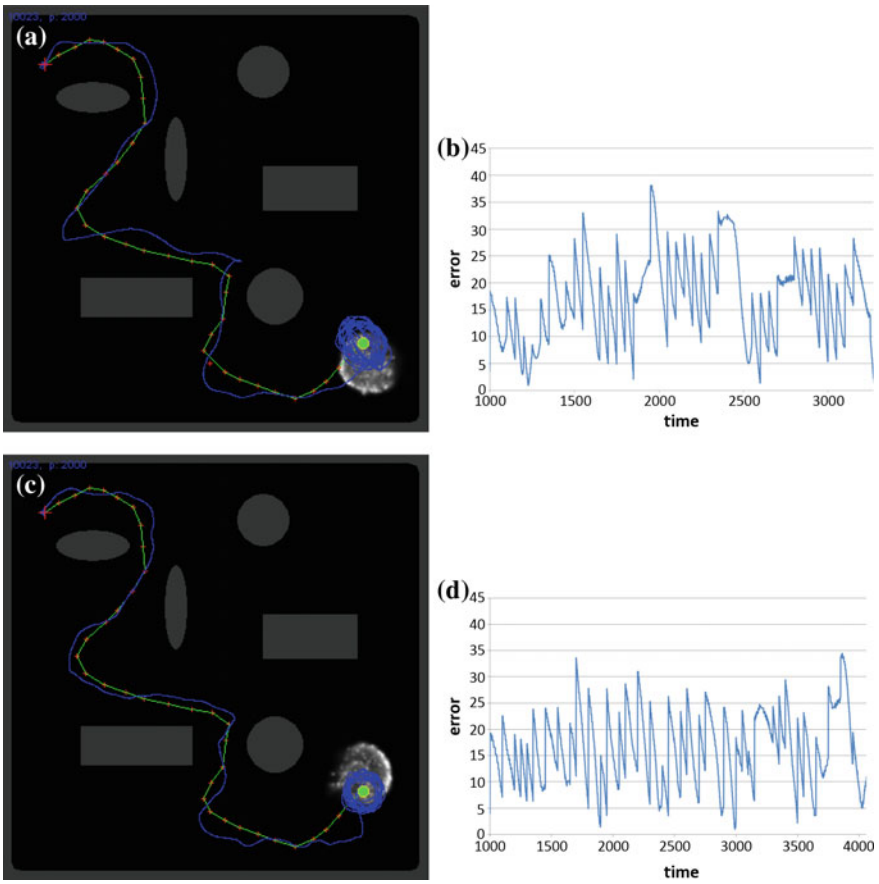

Fig. 21.10 Automated guidance of amorphous amoeboid robot via attractants. **a**, **c**, **e**, **g** Image showing trajectory of blob (*blue online*) as it is guided along path (*green online*) from start (*cross*) to finish (*circle*) with *pID* parameter values 0.001, 0.01, 0.05, 0.1 respectively, **b**, **d**, **f**, **h** plot showing error of blob position (in pixels) compared to path over time

path of the blob (blue, online) shows that the blob continues to traverse around the periphery of the final position. At low *pID* (high momentum) values, the radius of this circular movement is much larger than at high *pID* (low momentum) values.

As the blob traverses the points on the path, there is a repetitive sequence of error minimisation which occurs, as shown in Fig. 21.11 (which shows an enlarged portion of the migration plot of Fig. 21.10f between 2000–3000 steps). This is indicated by the 'sawtooth' profile of the plot. As the blob moves forward (attracted by the stimulus point presented at the next path node) the distance between the current centroid of the blob and the target node is minimised (the diminishing diagonal lines of the plot). When this distance is less than the distance between the current node, the new node is selected. The selection of the next node changes the stimulus point location and

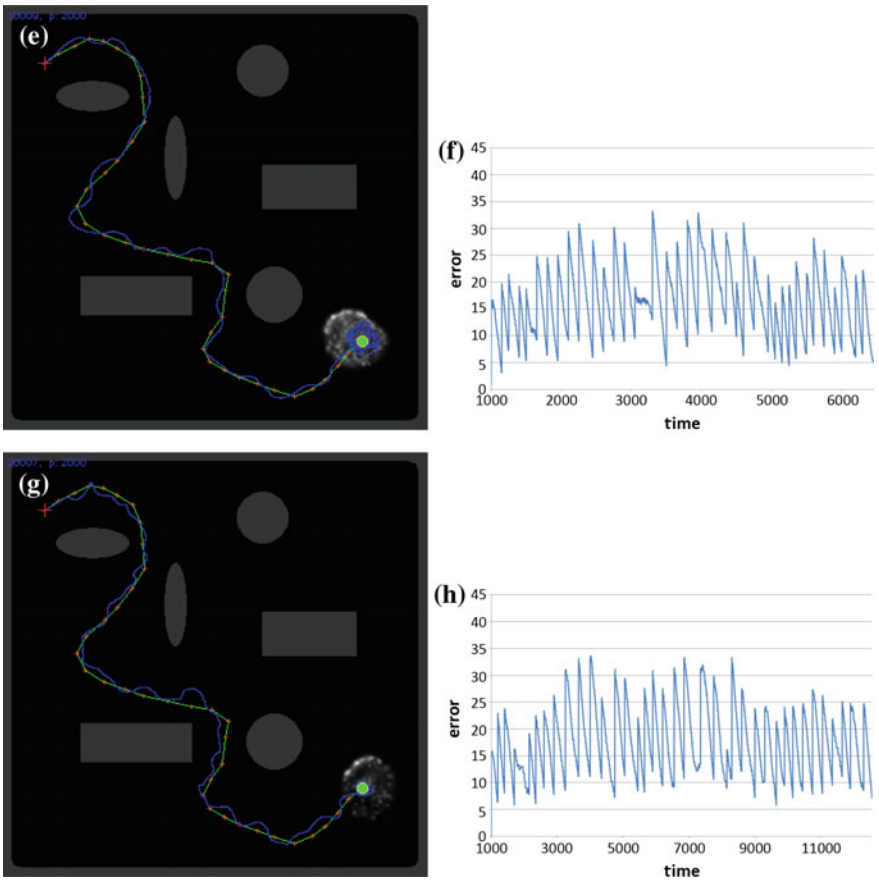

Fig. 21.10 (continued)

also causes a sudden jump in migration error (the vertical lines in the plot). This pattern of minimisation and new target selection occurs until the end node of the path is reached, at which point the blob will circle the final node on the path.

21.5.4 Automatic Guidance with Repellent Stimuli

As an alternative to pulling the blob towards the stimulus, it is also possible to 'push' the blob. This can be achieved by mimicking the response of slime mould to hazardous stimuli, for example exposure to light irradiation. In the face of such stimuli slime mould withdraws parts of its body plan away from exposed regions [40] and can thus be guided away from simple obstacles comprised of light-exposed areas [3]. In the model plasmodium we can reproduce the effect of exposure to light by altering

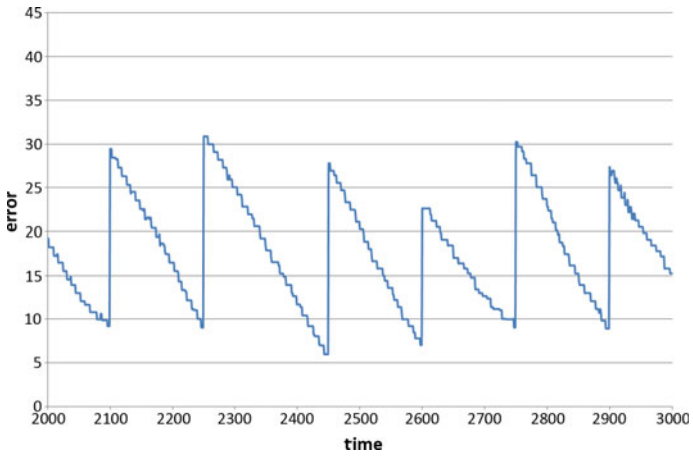

Fig. 21.11 Evolution of automated guidance during path traversal. Chart showing enlarged portion of Fig. 21.10f at time 2000–3000 steps. Sawtooth profile plot shows changes in migration error as the blob is guided from node to node along the path, minimising the distance error at the current node (*downward diagonal*), before the next node is selected, generating a new error distance (*vertical line*)

the sensitivity of particles exposed to illuminated regions of the lattice. This reduces flux within that region of the blob and also attraction to attractant stimuli located in these exposed regions. Due to the cohesion of the blob, the travelling waves moving within the blob are stronger in unilluminated regions and this propels the blob from exposed regions.

We can implement automated guidance by repellent stimuli by having the stimulus point represent an *absence* of illumination, for example a square masked region. *Outside* this region all other areas are temporarily exposed to simulated light exposure. This tends to maintain the blob within the confines of the masked region and also move peripheral parts of the blob that are outside of the protective masked region back inside the mask.

We tested the repellent method on the same obstacle arena as used in the attractant stimulus condition. Figure 21.12 shows examples of blob guidance through this arena at different *pID* values. As indicated by the chart plots showing distance from the chosen path, the general 'sawtooth' pattern of movement along the path is the same as in the attractant guidance method. However, the trajectory of the blob under light irradiation guidance shows much closer adherence to the original path and there is significantly less 'overshoot' than in the attractant guidance method when sudden changes in path direction occur. Across multiple runs of both attractant and repellent conditions the time taken to traverse the path was also shorter in the light-irradiation condition at all *pID* values (Fig. 21.13a). Furthermore, the mean error from the desired path was also lower for the light-irradiation condition, compared to the attractant guided method (Fig. 21.13b).

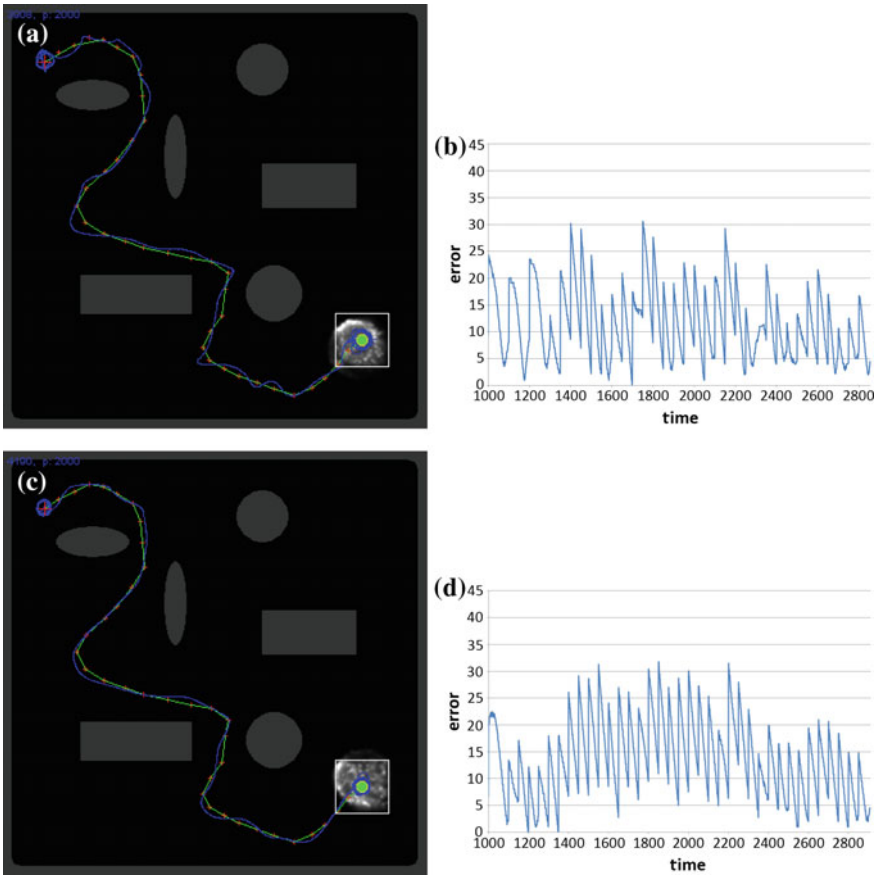

Fig. 21.12 Automated guidance of amorphous amoeboid robot via repellent stimulus of simulated light irradiation. **a**, **c**, **e**, **g** Image showing trajectory of blob (*blue online*) as it is guided along path (*green online*) from start (*cross*) to finish (*circle*) with *pID* parameter values 0.001, 0.01, 0.05, 0.1 respectively. Masked area is indicated by the *square* region surrounding the blob, **b**, **d**, **f**, **h** plot showing error of blob position (in pixels) compared to path over time

Why does the light irradiation guidance method track the path more accurately than the attractant method? The square masked region surrounding the blob (for example, Fig. 21.12a) illuminates all regions outside the mask (i.e. outside of the main blob region), providing simultaneous stimuli at different parts of the blob, compared to the single point stimulus in the attractant condition. Any particles in the illuminated region are subject to the reduction in flux and thus try to return to the unexposed region within the square. This suppression of flux outside the mask square has the effect of damping travelling waves outside the square, causing less momentum to build up, and more accurate traversal of the path.

It should be noted that although the high-momentum travelling waves in the attractant condition were responsible for the increased error from the desired path,

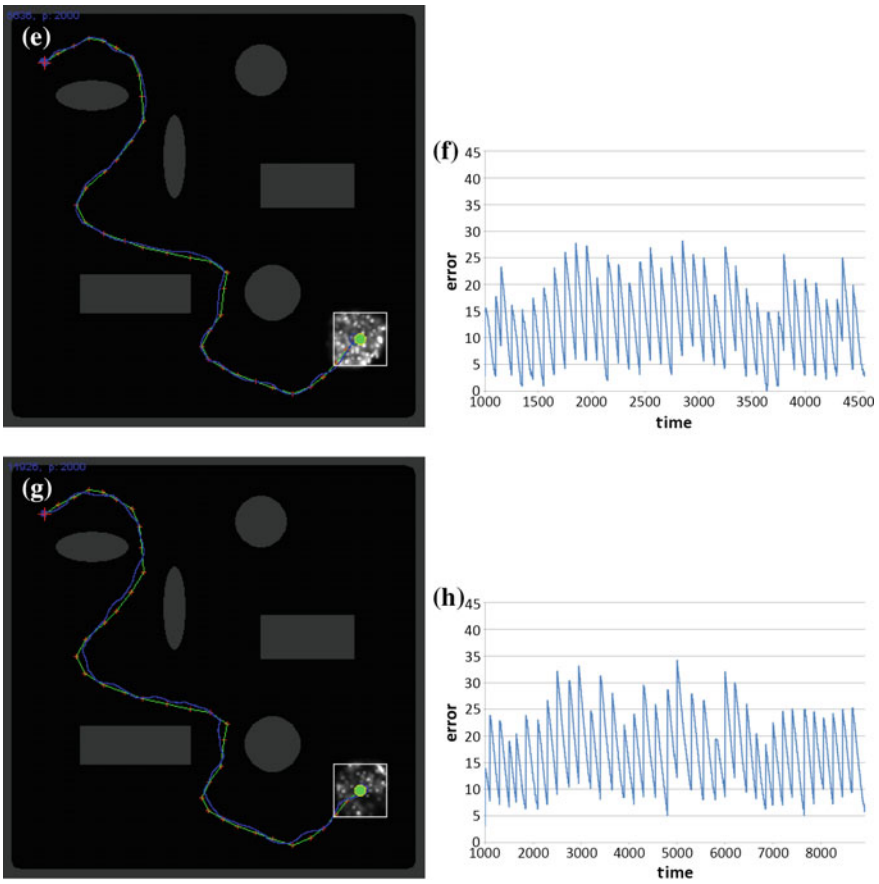

Fig. 21.12 (continued)

the oscillatory travelling waves are of critical importance for the blob movement. Indeed in the attractant stimulus method, the migration of the blob along the entire course could not be completed without oscillatory movement. In the light irradiation stimulus condition, the light mask was sufficient to move the non-oscillatory collective along the path, but the penalty was a greatly increased time of traversal — over 80,000 scheduler steps — compared to a range of 1900–8000 steps under oscillatory conditions.

21.5.5 Novel Properties of Guided Amoeboid Movement

In addition to the tracking abilities of the hybrid unconventional/conventional computing guidance methods, the amorphous and adaptive properties of the collective

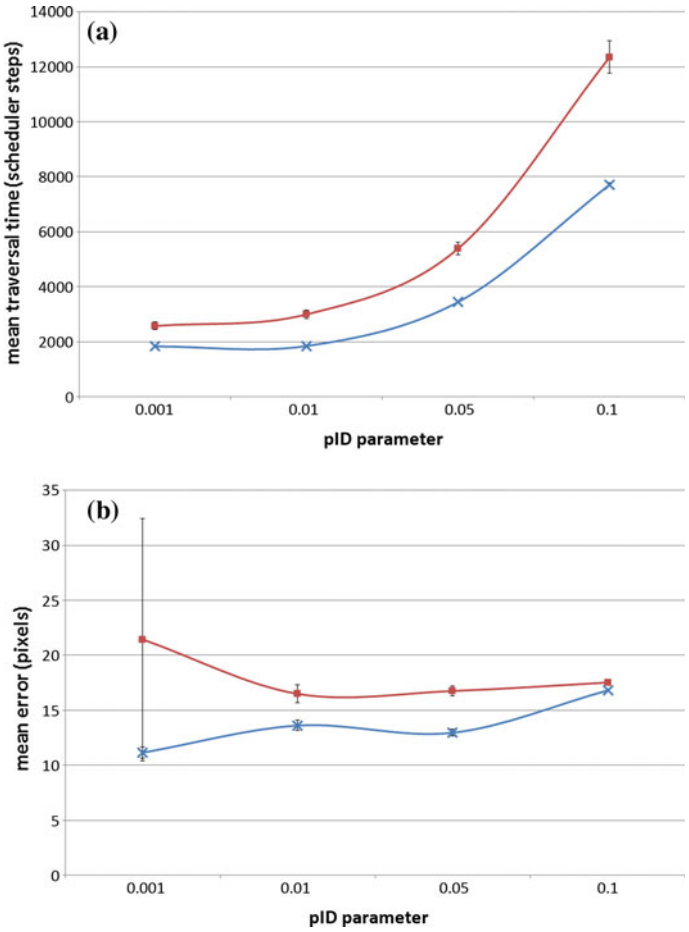

Fig. 21.13 Comparison of attractant and repellent guidance methods in terms of time and guidance errors. **a** comparison of path traversal time for different *pID* values for attractant (*squares*) and repellent (*crosses*) stimuli, **b** mean guidance error from path at different *pID* values for attractant (*squares*) and repellent (*crosses*) stimuli (mean of ten runs per *pID* value, standard deviation indicated). Guidance by repellent stimuli is faster and more accurate

result in some interesting properties during its movement. Figure 21.14 shows the guidance of the blob along a vertical arena (in this example by repellent light irradiation stimuli). The arena is composed of a narrow channel, some horizontal blocks and finally a very narrow grating, before the destination site (Fig. 21.14a). As the blob passes through the narrow channel, the blob elongates, adapting its shape automatically in order to fit through the narrow channel (Fig. 21.14b) before restoring its approximately circular shape once the channel has been crossed.

In the case of the grating at the bottom of the arena, the space between the grating obstacles is so narrow that the blob shape deforms dramatically in regions outside the

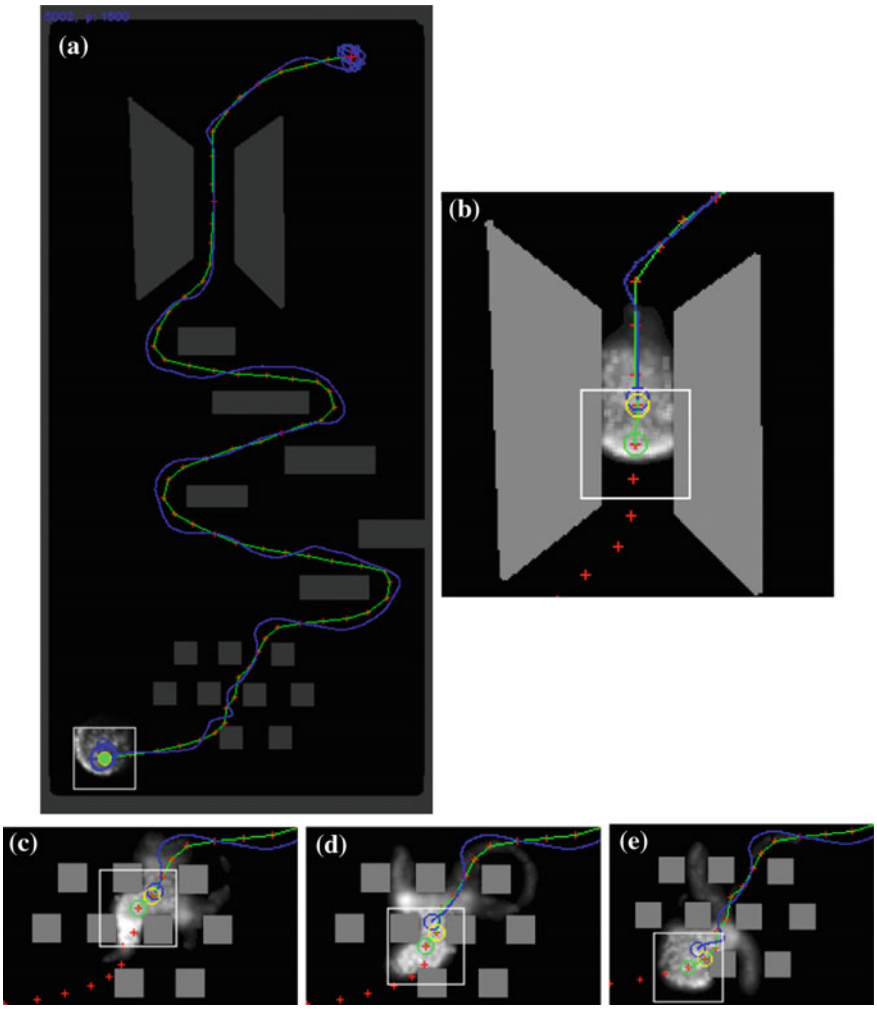

Fig. 21.14 Novel properties of the amoeboid blob as it navigates a complex vertical arena. **a** overview of traversed arena showing obstacles (*grey*), path (*green*, *online*), and blob trajectory (*blue*, *online*), **b** blob elongates as it passes through a narrow tunnel, **c**–**e** blob is distorted as it is forced through a narrow grating by the stimulus mask before re-forming its shape when the obstacle is passed

mask, forming writhing pseudopodium-like tendrils (Fig. 21.14c–e). Again, once the grating has been crossed the blob reforms its shape as it moves to the goal site. Despite this significant distortion of blob shape the path taken by the blob is fairly close to the target path (Fig. 21.14e). These properties are a function of the unconventional computing part of the system, in that they arise as an emergent property of the low-level particle interactions.

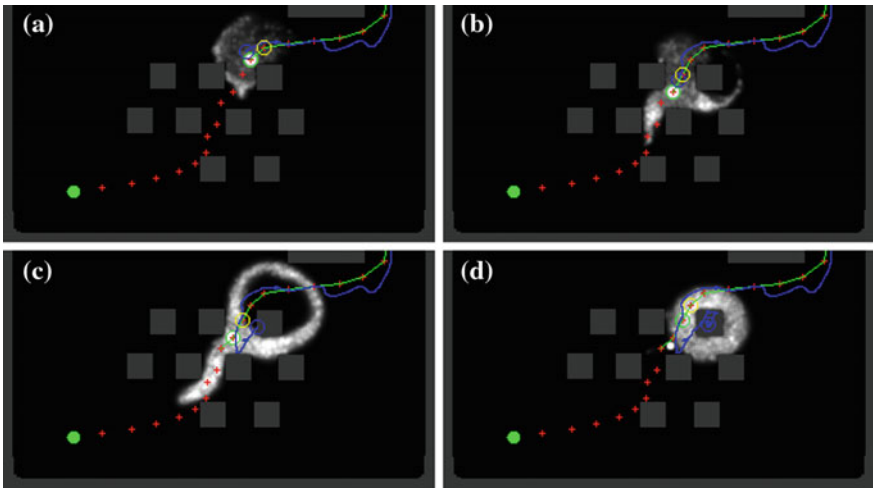

Fig. 21.15 Guided blob becomes stuck on grating using attractant method. **a** blob enters grating area, **b** distortion of blob pattern occurs, **c** blob becomes entwined on grating obstacle, **d** blob minimises around grating and continues to cycle in this position

The distortion and re-formation of the blob shape at the narrow grating does not occur reliably with the attractant based guidance method, however. Figure 21.15 shows a blob guided by a single attractant source entering the grating region (Fig. 21.15a) where its body plan is distorted on contact with the obstacles (Fig. 21.15b). The blob becomes entwined on a single obstacle in the grating (Fig. 21.15c) and minimises its shape to wrap around the obstacle (Fig. 21.15d). The blob remains in this position indefinitely, cycling around this obstacle. Corruption of the X and Y stimulus values with Gaussian noise (to try to present multiple stimulus sites) does not detach the blob from the obstacle. This behaviour again demonstrates the effectiveness of guidance by illumination masked regions compared to the attractant guidance method. Why does the blob not become stuck at this obstacle when guided by the repellent mask? Again, this is because the illumination mask presents multiple stimulus points to the blob (at the interface of the square mask edges which contact the blob), whereas the attractant guidance method only presents a single guidance stimulus.

21.5.6 Emergency Recovery Mode for Lost Collectives

In over 80 experiments with the guidance mechanisms we only encountered one instance (attractant stimulus condition, with low *pID* 0.001) where the blob migrated far from the desired path and was not influenced by the presented attractant stimulus (incidentally, this was the cause for the large deviation bars in the first data point of the

attractant series in Fig. 21.13b). In this instance the 'lost' blob spontaneously moved (via random migration) back near to the path where it once again was influenced by the presented stimuli. However, this led us to devise a 'recovery mode' to cater for instances when the amoeboid blobs may lose sight of the path. This mechanism was implemented in the following way (Fig. 21.16). At each scheduled comparison of blob centroid position and the closest nearest path position, the Euclidean distance (in pixels) was compared to a threshold value θ. If θ (set to 100 pixels) was exceeded a binary *lost* flag was set and an attractant stimulus (or light illumination mask for the repellent condition) was projected at random locations along an imaginary line from the blob centroid to the nearest path point (Fig. 21.16b). These additional stimuli attract the blob and continue to be presented until the distance from the blob to the path point was $< \theta$, at which time the *lost* flag was reset and normal guidance resumed. This mechanism is sufficient to guide the errant blob back onto the correct path.

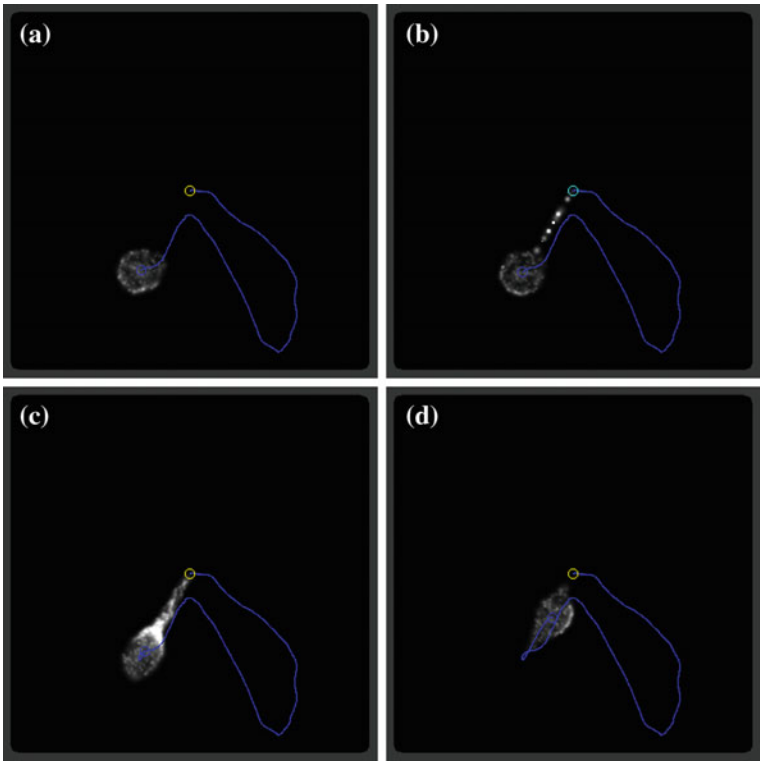

Fig. 21.16 Automated recovery of 'lost' blob. **a** blob migrates away from centre position and exceeds θ, triggering recovery mode, **b** a stream of attractant stimuli is presented between the blob and the target point, **c** the blob is attracted to the stimuli and starts to migrate back to the target, **d** recovery stimuli are halted when the distance $< \theta$

21.6 Discussion

Control and guidance of collective soft-robotics devices is a very challenging problem for classical computing when the robotic devices themselves harness emergent behaviours arising from the local component interactions of the collective itself. In this article we have examined the problem of generating and controlling collective amoeboid movement from an unconventional computing perspective using a multi-agent model of slime mould *Physarum polycephalum*. Collective movement was generated in a morphologically adaptive 'blob' comprising a population of simple particles on a diffusive lattice. Taking inspiration from slime mould, the position of this blob could be altered by the spatial placement of attractant and repellent stimuli. Simple open-loop mechanisms were demonstrated using attractant stimuli which enabled automatic, but uncontrolled, movement along a pre-defined path. In order to automatically guide the movement of the multi-agent collective, however, a hybrid approach utilising features of unconventional and classical computation was required.

The hybrid approach utilised the self-organised generation of blob cohesion and its movement by oscillatory travelling waves (the unconventional computing substrate) in a blob of fixed population size. This was combined with a classically implemented closed-loop mechanism to ascertain the current position of the blob in relation to a pre-defined path. By comparing the current blob position with the closest point on the path, a stimulus (the next available path point) was then presented to the unconventional computing substrate, causing the blob to migrate along the path. Of the two stimulus types investigated (attractant and repellent) to guide the blob, the repellent stimulus (masking the blob from simulated light irradiation) resulted in a faster path traversal with fewer errors (in terms of distance from the pre-defined path) and allowed the blob to pass automatically through very narrow gratings. Passage through a narrow grating could not be achieved using the attractant stimulus condition, due to the lack of simultaneous stimuli to the blob when compared to the repellent mask method. The momentum of the blob could be controlled by adjusting a parameter of the model. Stronger momentum resulted in faster path traversal in both stimulus types, but resulted in a characteristic overshooting of path corners in the attractant stimulus condition.

This hybrid approach successfully combines classical computing (tracking and stimulus location) with unconventional computing (generation of self-organised travelling waves to generate amoeboid movement). The unconventional computing part of the system accounts for novel properties of the amoeboid robot such as cohesion, spontaneous oscillatory movement and automatic deformation of shape (and subsequent re-formation) in the presence of obstacles. The result is a parsimonious combination of classical computing and its benefits (for example, rapid and efficient arithmetic calculations for tracking and guidance), combined with the desirable features provided by unconventional computing (self-organised movement, resilience to deformation) that are a natural fit for unconventional computing substrates and which would be difficult to implement in a classical system. The result of this hybrid

approach is a system which combines the best features of both approaches. We hope that this work will provide a useful contribution towards future implementations of soft-bodied robotic systems which utilise hybrid unconventional and classical computing methods.

Acknowledgments This research was supported by the EU research project "Physarum Chip: Growing Computers from Slime Mould" (FP7 ICT Ref 316366).

References

1. Adamatzky, A.: Physarum machines: encapsulating reaction-diffusion to compute spanning tree. Naturwissenschaften **94**(12), 975–980 (2007)
2. Adamatzky, A.: Developing proximity graphs by Physarum polycephalum: does the plasmodium follow the toussaint hierarchy. Parallel Processing Letters **19**, 105–127 (2008)
3. Adamatzky, A.: Steering Todium with Light: Dynamical Programming of *Physarum* Machine (2009). arXiv:0908.0850
4. Adamatzky, A.: Manipulating substances with Physarum polycephalum. Mater. Sci. Eng. C **38**(8), 1211–1220 (2010)
5. Adamatzky, A.: Slime mould tactile sensor. Sens. actuators B: Chem. **188**, 38–44 (2013)
6. Adamatzky, A., de Lacy Costello, B., Melhuish, C., Ratcliffe, N.: Experimental reaction-diffusion chemical processors for robot path planning. J. Intell. Robot. Syst. **37**(3), 233–249 (2003)
7. Adamatzky, A., de Lacy, Costello, B., Shirakawa, T.: Universal computation with limited resources: Belousov-Zhabotinsky and Physarum computers. Int. J. Bifurc. Chaos **18**(8), 2373–2389 (2008)
8. Adamatzky, A., Jones, J.: Towards Physarum robots: computing and manipulating on water surface. J. Bionic Eng. **5**(4), 348–357 (2008)
9. Brugués, A., Anon, E., Conte, V., Veldhuis, J.H., Gupta, M., Colombelli, J., Muñoz, J.J., Brodland, G.W., Ladoux, B., Trepat, X.: Forces driving epithelial wound healing. Nat. Phys. (2014)
10. Buhl, J., Sumpter, D.J.T., Couzin, I.D., Hale, J.J., Despland, E., Miller, E.R., Simpson, S.J.: From disorder to order in marching locusts. Science **312**(5778), 1402 (2006)
11. Carlile, M.J.: Nutrition and chemotaxis in the myxomycete Physarum polycephalum: the effect of carbohydrates on the plasmodium. J. Gen. Microbiol. **63**(2), 221–226 (1970)
12. Durham, A.C.H., Ridgway, E.B.: Control of chemotaxis in Physarum polycephalum. J. Cell Biol. **69**, 218–223 (1976)
13. Friedl, P., Locker, J., Sahai, E., Segall, J.E.: Classifying collective cancer cell invasion. Nat. Cell Biol. **14**(8), 777–783 (2012)
14. Gale, E., Adamatzky, A., de Lacy Costello, B.: Are slime moulds living memristors? Technical report (2013)
15. Garnier, S., Gautrais, J., Theraulaz, G.: The biological principles of swarm intelligence. Swarm Intell. **1**(1), 3–31 (2007)
16. Gholami, A., Steinbock, O., Zykov, V., Bodenschatz, E.: Flow-driven waves during pattern formation of Dictyostelium discoideum. Bull. Am. Phys. Soc. **60** (2015)
17. Gov, N.S.: Collective cell migration. Cell Matrix Mech. 219 (2014)
18. Gueron, S., Levin, S.A.: Self-organization of front patterns in large wildebeest herds. J. Theor. Biol. **165**(4), 541–552 (1993)
19. Helbing, D.: Traffic and related self-driven many-particle systems. Rev. Mod. Phys. **73**(4), 1067 (2001)

20. Helbing, D., Johansson, A.: Eidgenössische Technische Hochschule. Pedestrian, crowd and evacuation dynamics. Swiss Federal Institute of Technology (2009)
21. Helbing, D., Molnar, P., Farkas, I.J., Bolay, K.: Self-organizing pedestrian movement. Environ. Plan. B **28**(3), 361–384 (2001)
22. Herbert-Read, J.E., Perna, A., Mann, R.P., Schaerf, T.M., Sumpter, D.J.T., Ward, A.J.W.: Inferring the rules of interaction of shoaling fish. Proc. Natl. Acad. Sci. **108**(46), 18726–18731 (2011)
23. Jones, J.: Characteristics of pattern formation and evolution in approximations of Physarum transport networks. Artif. Life **16**(2), 127–153 (2010)
24. Jones, J.: The emergence and dynamical evolution of complex transport networks from simple low-level behaviours. Int. J. Unconv. Comput. **6**, 125–144 (2010)
25. Jones, J.: From Pattern Formation to Material Computation: Multi-agent Modelling of Physarum Polycephalum. Springer, Heidelberg (2015)
26. Jones, J.: Mechanisms inducing parallel computation in a model of Physarum polycephalum transport networks. Parallel Process. Lett. **25**(01), 1540004 (2015)
27. Jones, J.: A morphological adaptation approach to path planning inspired by slime mould. Int. J. Gen. Syst. **44**(3), 279–291 (2015)
28. Jones, J., Adamatzky, A.: Emergence of self-organized amoeboid movement in a multi-agent approximation of Physarum polycephalum. Bioinspiration Biomim. **7**(1), 016009 (2012)
29. Jones, J., Tsuda, S., Adamatzky, A.: Towards *Physarum* robots. In: Bio-Inspired Self-Organizing Robotic Systems, pp. 215–251 (2011)
30. Kalogeiton, V.S., Papadopoulous, D.P., Sirakoulis, G.Ch., Vazquez-Otero, A., Faigl, J., Duro, N., Dormido, R., Henderson, T.C., Joshi, A., Rashkeev, K. et al.: Hey *Physarum*! can you perform slam? Int. J. Unconv. Comput. **10**(4) (2014)
31. Kessler, M.A., Werner, B.T.: Self-organization of sorted patterned ground. Science **299**(5605), 380–383 (2003)
32. Kishimoto, U.: Rhythmicity in the protoplasmic streaming of a slime mould, Physarum polycephalum. J. Gen. Physiol. **41**(6), 1223–1244 (1958)
33. Kocurek, G., Ewing, R.C.: Aeolian dune field self-organization-implications for the formation of simple versus complex dune-field patterns. Geomorphology **72**(1), 94–105 (2005)
34. Lee, K.J., Cox, E.C., Goldstein, R.E.: Competing patterns of signaling activity in Dictyostelium discoideum. Phys. Rev. Lett. **76**(7), 1174 (1996)
35. Levin, M.: Morphogenetic fields in embryogenesis, regeneration, and cancer: non-local control of complex patterning. Biosystems **109**(3), 243–261 (2012)
36. Matsushita, M., Wakita, J., Itoh, H., Watanabe, K., Arai, T., Matsuyama, T., Sakaguchi, H., Mimura, M.: Formation of colony patterns by a bacterial cell population. Physica A: Stat. Mech. Appl. **274**(1), 190–199 (1999)
37. Nagatani, T.: The physics of traffic jams. Rep. Progr. Phys. **65**(9), 1331 (2002)
38. Nakagaki, T., Ueda, T.: Phase switching of rhythmic contraction in relation to regulation of amoeboid behavior by the plasmodium of Physarum polycephalum. J. Theor. Biol. **179**, 261–267 (1996)
39. Nakagaki, T., Yamada, H., Toth, A.: Intelligence: Maze-solving by an amoeboid organism. Nature **407**, 470 (2000)
40. Nakagaki, T., Yamada, H., Ueda, T.: Modulation of cellular rhythm and photoavoidance by oscillatory irradiation in the Physarum plasmodium. Biophys. Chem. **82**(1), 23–28 (1999)
41. Nakagaki, T., Yamadaa, H., Ueda, T.: Interaction between cell shape and contraction pattern in the physarum plasmodium. Biophys. Chem. **84**, 195–204 (2000)
42. Parrish, J.K., Turchin, P.: Individual decisions, traffic rules, and emergent pattern in schooling fish. In: Animal Groups in Three Dimensions, pp. 126–142 (1997)
43. Rauprich, O., Matsushita, M., Weijer, C.J., Siegert, F., Esipov, S.E., Shapiro, J.A.: Periodic phenomena in Proteus mirabilis swarm colony development. J. Bacteriol. **178**(22), 6525–6538 (1996)
44. Reynolds, C.W.: Flocks, herds and schools: A distributed behavioral model. In: ACM SIGGRAPH Computer Graphics, vol. 21, pp. 25–34. ACM, New York (1987)

45. Saigusa, T., Tero, A., Nakagaki, T., Kuramoto, Y.: Amoebae anticipate periodic events. Phys. Rev. Lett. **100**(1) (2008)
46. Serini, G., Ambrosi, D., Giraudo, E., Gamba, A., Preziosi, L., Bussolino, F.: Modeling the early stages of vascular network assembly. EMBO J. **22**(8), 1771–1779 (2003)
47. Shirakawa, T., Adamatzky, A., Gunji, Y.-P., Miyake, Y.: On simultaneous construction of Voronoi diagram and Delaunay triangulation by Physarum polycephalum. Int. J. Bifurc. Chaos **19**(9), 3109–3117 (2009)
48. Shirakawa, T., Gunji, Y.-P.: Computation of Voronoi diagram and collision-free path using the Plasmodium of Physarum polycephalum. Int. J. Unconv. Comput. **6**(2), 79–88 (2010)
49. Shirakawa, T., Konagano, R., Inoue, K.: Novel taxis of the *Physarum* plasmodium and a taxis-based simulation of *Physarum* swarm. In: 2012 Joint 6th International Conference on Soft Computing and Intelligent Systems (SCIS) and 13th International Symposium on Advanced Intelligent Systems (ISIS), pp. 296–300. IEEE, New York (2012)
50. Stephenson, S.L., Stempen, H., Hall, I.: Myxomycetes: A Handbook of Slime Molds. Timber Press Portland, Oregon (1994)
51. Sumpter, D.J.T.: Collective Animal Behavior. Princeton University Press, Princeton (2010)
52. Takagi, S., Ueda, T.: Emergence and transitions of dynamic patterns of thickness oscillation of the plasmodium of the true slime mold Physarum polycephalum. Physica D **237**, 420–427 (2008)
53. Takamatsu, A., Fujii, T.: Time delay effect in a living coupled oscillator system with the plasmodium of Physarum polycephalum. Phys. Rev. Lett. **85**, 2026–2029 (2000)
54. Takamatsu, A., Takaba, E., Takizawa, G.: Environment-dependent morphology in plasmodium of true slime mold Physarum polycephalum and a network growth model. J. Theor. Biol. **256**(1), 29–44 (2009)
55. Tsuda, S., Aono, M., Gunji, Y.-P.: Robust and emergent Physarum logical-computing. BioSystems **73**, 45–55 (2004)
56. Tsuda, S., Jones, J.: The emergence of synchronization behavior in Physarum polycephalum and its particle approximation. Biosystems **103**, 331–341 (2010)
57. Tsuda, S., Jones, J., Adamatzky, A.: Towards Physarum engines. Appl. Bionics Biomech. **9**(3), 221–240 (2012)
58. Tsuda, S., Zauner, K.-P., Gunji, Y.-P.: Robot control with biological cells. BioSystems **87**, 215–223 (2007)
59. Ueda, T., Terayama, K., Kurihara, K., Kobatake, Y.: Threshold phenomena in chemoreception and taxis in slime mold Physarum polycephalum. J. Gen. Physiol. **65**(2), 223–234 (1975). February
60. Umedachi, T., Kitamura, T., Takeda, K., Nakagaki, T., Kobayashi, R., Ishiguro, A.: A modular robot driven by protoplasmic streaming. Distrib. Autonom. Robot. Syst. **8**, 193–202 (2009)
61. Vicsek, T., Zafeiris, A.: Collective motion. Phys. Rep. **517**, 71–140 (2012)
62. Werner, B.T.: Eolian dunes: computer simulations and attractor interpretation. Geology **23**(12), 1107–1110 (1995)
63. Whiting, J.G.H., de Lacy Costello, B.P.J., Adamatzky, A.: Slime mould logic gates based on frequency changes of electrical potential oscillation. Biosystems **124**, 21–25 (2014)
64. Whiting, J.G.H., de Lacy Costello, B.P.J., Adamatzky, A.: Towards slime mould chemical sensor: Mapping chemical inputs onto electrical potential dynamics of Physarum Polycephalum. Sens. Actuators B: Chem. **191**, 844–853 (2014)
65. Wolf, R., Niemuth, J., Sauer, H.: Thermotaxis and protoplasmic oscillations in Physarum plasmodia analysed in a novel device generating stable linear temperature gradients. Protoplasma **197**(1–2), 121–131 (1997)
66. Wolke, A., Niemeyer, F., Achenbach, F.: Geotactic behavior of the acellular myxomycete Physarum polycephalum. Cell Biol. Int. Rep. **11**(7), 525–528 (1987)
67. Zheng, X., Zhong, T., Liu, M.: Modeling crowd evacuation of a building based on seven methodological approaches. Build. Environ. **44**(3), 437–445 (2009)