

A Multi-scaled Approach to Artificial Life Simulation With P Systems and Dissipative Particle Dynamics

James Smaldon
School of Computer Science
University of Nottingham
Jubilee Campus, Nottingham,
NG8 1BB, UK
jqs@cs.nott.ac.uk

Jonathan Blakes
School of Computer Science
University of Nottingham
Jubilee Campus, Nottingham,
NG8 1BB, UK
jvb@cs.nott.ac.uk

Dr. Natalio Krasnogor
School of Computer Science
University of Nottingham
Jubilee Campus, Nottingham,
NG8 1BB, UK
nxk@cs.nott.ac.uk

Prof. Doron Lancet
Department of Molecular
Genetics
Weizmann Institute of
Science, Rehovot, Israel
doron.lancet@weizmann.ac.il

ABSTRACT

Compartmentalisation is thought to have been a crucial step in the origin of life. To help us bridge the gap between self-assembly processes behind the formation of bio-compartments and metabolic and information bearing processes we refer to DPD and P Systems Simulations. In this paper we outline a new software platform linking a high level abstract computational formalism (P Systems) with a molecular scale model (Dissipative Particle Dynamics) by linking the membranes which delimit the cellular regions within P Systems to self-assembled phospholipid based vesicles in DPD. We test the platform by modelling a passive transport process involving vesicles containing membrane inclusions similar to pore complexes such as α -hemolysin. In doing so, we illustrate the usefulness of the modelling approach and derive a more realistic parameter set for the P system through the dissipative particle dynamics simulation.

Categories and Subject Descriptors

I.6 [Simulation and Modelling]: Model Development;
I.6 [Simulation and Modelling]: Types of Simulation—
Combined

General Terms

Experimentation, Verification, Algorithms

Keywords

Artificial Life, Synthetic Biology, Systems Biology, P Systems, Dissipative Particle Dynamics

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. To copy otherwise, to republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee.

GECCO'08, July 12–16, 2008, Atlanta, Georgia, USA.
Copyright 2008 ACM 978-1-60558-130-9/08/07 ...\$5.00.

1. INTRODUCTION

Research in ALife has always been tightly coupled to advances in complex systems, which are understood here as systems where a large number of entities interacting repetitively and (usually) far from equilibrium. Complex systems analysis has traditionally relied on systems of differential equations, statistical mechanics and other formal techniques to represent some aggregated average of the behaviour that emerges when the entities interact. In contrast, the bulk of ALife research in the last 20 years has been done with a more ad-hoc bottom-up engineering approach by designing or evolving the rules that govern the local interactions of the entities in the system as to produce certain emergent behaviour. Emergence in this context is interpreted as a process within the system that could not have been predicted from merely inspecting the rules but that it is observed only by running the simulation (for early examples of flocking and herding emergent behaviour see [24]).

Some of the earliest landmarks in ALife (T. Rays Tierra for example) were based on an individual based modelling framework, which were highly abstract and quite limited in the simulated details (i.e. physical and chemical laws) of the environment where the agents performed their interactions. K. Sims virtual creatures incorporated a more accurate (albeit still arbitrary) physical reality into the ALife system. In turn, this increase in the detail of the environmental interactions allowed richer emergent processes to be observed. Later work incorporated a more detailed biology through the addition of developmental processes, differential gene expression and genetic regulatory networks endowing ALife simulations with greater realism [5, 7]. In the same vein, the Graded Autocatalysis Replication Domain (GARD) model has been used to simulate the behaviour of an ensemble of mutually catalytic molecules, resulting in emergent properties such as self reproduction with mutations[18, 28, 29, 30]. Thus, as computing resources became more accessible and our biological knowledge deepened, more and more levels of physical, chemical and biological details were included in a bottom-up fashion into ALife simulations[15].

Recent advances in analytical biotechnology, computational biology, bioinformatics and microbiology are trans-

forming our views of the complexity of biological systems, particularly the computations they perform (i.e. how information is processed, transmitted and stored) in order to survive, adapt and evolve in dynamic and sometimes hostile environments. In this paper we capture some of these more recent biological insights, in particular those related to cell biology, as to develop sophisticated ALife simulations of cellular-like systems. Furthermore, while we propose to stick to the traditional engineering approach of building ALife systems from the bottom-up we would like to extend current research practice towards a more computationally formal and rigorous approach to the design and implementation of ALife research.

For this purpose we present a software platform based on P system modelling and DPD simulations. The former is used to formally capture what are universally accepted to be the three key required functional components of a minimal cell[23], namely, compartments, metabolism and information polymers (e.g. DNA, RNA, proteins, etc). The latter is used to model bottom up the self assembly of the compartments formally represented in the P systems. We demonstrate that it is possible, in principle, to bridge the gap between a rigorous mathematical formalism for cell-like systems based on P systems and the self-assembly of compartments based on DPD simulations. At a later stage, we plan to introduce into this platform features based on the GARD model, such as compositional inheritance, components and clustering-based compotypes[28, 29, 30].

2. REPRODUCING THE KEY ELEMENTS OF LIFE IN SIMULATION

Biological life is immensely complex and varied, and this makes a universal definition difficult. To avoid such a definition, D. Harel[12] and Cronin et al. [4] recently proposed a test similar to the Turing test for intelligence than can be applied to artificial life. However, certain properties can be thought of as being integral to much of the cellular based life of the planet. Three of these properties in particular have been of interest in synthetic biology (when applying the bottom-up design approach) and protolife research[22]. These are encapsulation or separation of cellular sub-volumes within containers, a metabolism which involves the refining, extraction and storage of energy from the surrounding medium and some form of information transfer, capturing and propagating the physical manifestation of the life form over much longer time scales than the life of single organism. We next discuss in more detail each of these key components.

2.1 The Container

The importance of a container demarcating the artificial cells physical boundary between self and non-self was recognised [19] by the very first investigation into the origins of life research community as being an essential experimental distinction rather than only a philosophical axiom. Key contemporary studies [17, 25, 22] have also emphasised the crucial role of the container as a barrier that regulates (e.g., controls energy and raw material flow such that artificial cell may be sustained far from equilibrium) and protects (e.g., controls the dissipation of information molecules associated with the identity of the artificial cell) other functional components of the cell.

2.2 Metabolism

The primarily spatial control of the container is juxtaposed by the primarily temporal control of metabolism. It is likely the most context dependent of the three building blocks since whilst an artificial cell might successfully use its own information codes locked within its own suitable functionalised containment, the fuelling of the artificial cell will depend on what is available from the environment. The metabolism harvests raw materials from the environment, a process that is dictated by local availability and stoichiometry but that may manipulate concentration or catalysis processes as to achieve the functional aims of the artificial life form. One of the major goals of artificial life endeavours is to show how metabolism can give rise to container construction[28, 29].

2.3 Information

Our traditional views of conventional life often encourages focus upon life codes, the most celebrated being that conveyed by nucleic acids. The forms of information and the methods by which it propagates may, however, be varied.

The concept of the survival of a given unit of information is intimately tied up with the notion of life. The ability of one gene to confer fitness upon the organism that carries it, is the primary mode of selection and evolution.

Although nucleic acids are the most pervasive method of information storage and transfer in cells, information may be present in other forms, not only in a discrete set of coding molecules but also in e.g., the time-averaged position of one functional container molecule with respect to another. For example in [3] Cairns-Smith theorised that evolutionary processes could occur in crystals by propagations of imperfections in the crystalline structure. Another notable example is the storage and transmission of compositional information, as manifested in the GARD model. Here, it is the counts of molecules within a protocellular entity, rather than the sequence of subunits in an informational biopolymer, that is being transmitted to progeny [28, 29, 31]. This is in analogy to the epigenetic transmission of maternal mRNA counts to a daughter cell.

3. SIMULATION METHODOLOGY

In the rest of this paper we describe a simulation methodology which bridges the gap between an abstract formalism including explicitly the three elements of life described above, namely containers, metabolism and information processing, (P Systems) and a molecular scale simulation (DPD). In describing this methodology, we aim to create a multi-scale design and simulation tool for artificial life, enabling models to be expressed and prototyped in the abstract and biologically intuitive formalism, and then verified in DPD. By simulating the system in more detail, a concordance between the P System and DPD simulation results can be found, enabling a more realistic parameterisation of the P systems model. As many extensions to P systems have been shown to be Turing complete [21] the formalism is a syntax that enables construction of schema that are simultaneously concise and biologically apt, while also being highly computable. Once the P system model can be shown to correlate well with the results from the DPD simulations (which are limited in terms of time scale), it will be possible to perform

extended simulation and modelling with the constrained P system. Also, membrane processes which are implicit within the current formalism can be investigated explicitly. Thus conceptual and practical refinements at both ends of the scale are, in principle, possible.

3.1 P Systems

P Systems are a nature inspired computational paradigm, first proposed by Gh. Păun[21], that includes explicitly the three key properties of life described above. If a connection between P Systems and *in vivo* processes can be made, then P systems can be used as a concise and biologically relevant design tool for Alife systems, synthetic biology and protolife research. Indication that this is indeed possible comes from recent studies of P systems for systems biology[1, 2].

In this formalism, membranes are the containers, each delimiting a region containing a multiset of objects representing molecules and potentially other membranes into which objects can be communicated. These multisets are subject to rewriting rules governing the evolution of the system. The application of rules is analogous to a metabolism where some objects are consumed to create others, and can therefore model complexation, gene transcription, translation and post-translational modifications discretely.

Regions and rules are mapped to membranes using labels which determine their initial configuration and subdivide functionality within the system. Information is embodied in the state of the P system: the current membrane hierarchy, object multiplicity and rule applicability determine its next potential transition.

A P system of (degree m) is formally defined as:

$$\Pi = (O, \mu, w_1, w_2, \dots, w_m, R_1, R_2, \dots, R_m),$$

- O is a finite and non-empty alphabet of *objects*;
- μ is a *membrane structure* containing m membranes indexed $1, 2, \dots, m$. The membrane structure can be represented formally as a rooted tree but can also be depicted as a non-intersecting Venn diagram or nested pairs of indexed square brackets;
- w_1, w_2, \dots, w_m are the initial *multisets* of objects present in regions $1, 2, \dots, m$ of the membrane structure, written in the form a^3b^2c ;
- R_1, R_2, \dots, R_m are finite sets of rewriting rules associated with regions $1, 2, \dots, m$, which in the simplest¹ form permitting communication of objects between membranes are written:

$$u \xrightarrow{c} (v, tar) \quad (1)$$

where u is a multiset of objects from O and $v \in (O \times Tar)$, where $Tar = \{here, in, out\}$; *here* is the region with which the rule is associated, *out* the ascendent region or the environment in the case of the outer membrane, *in* a non-deterministically chosen descendent membrane. The importance of the constant c is discussed below.

The application of rules in a P system was originally performed in a non-deterministic and maximally-parallel manner, where any rule that at each step can be applied to the

¹More complex rule forms and special operators such as membrane dissolution that extend the functionality of P systems exist in the literature but are not required here.

system is chosen randomly with equal probability and applied exhaustively to the multiset in its associated region, not unlike L-Systems. The system evolves in uniform steps until no further rules can be applied. Processes such as communication of objects between regions occur instantly and without regard for concentration gradients or membrane permeability. Such an execution model is entirely incompatible with modelling of biological and chemical systems where reactions occur individually and on time scales many orders of magnitude removed.

To move towards a more realistic simulation with P systems, we employ a version of Gillespie's well known Stochastic Simulation Algorithm (SSA) [8] modified for the multi-compartmental structure of P systems. The stochastic rate constant c associated with rules of the form (1) is used to compute the *propensity* of a rule which, in proportion to the availability and combinatorial product of reactants, computes a suitable time to wait before that rule can be applied and so schedules rule application following Gillespie's theory of stochastic kinetics. For second-order reactions:

$$propensity = \begin{cases} c |a||b| & a \neq b \\ c \frac{|a|(|a|-1)}{2} & a = b \end{cases}$$

Despite the usefulness of the P Systems formalism as an abstract computational device[21], it is not clear if the interactions involving membranes or the membrane structures themselves are models of entities that are feasible *in vitro*. In P systems, regions have no spatial element, and complex processes such as membrane fission and fusion are conceptualised as atomic operations, rather than complex processes in their own right. Because of this reduction in the level of detail (which is necessary to produce a concise computational formalism) it is desirable to know whether models created in P systems maintain a connection with biological or chemical cells, or if the models are too abstract to effectively analyse models of biochemical systems.

In order to evaluate the accuracy of a stochastic P system description, we propose a mapping between the high level abstract formalism and a low level molecular scale particle dynamics model, namely Dissipative Particle Dynamics (DPD). By expressing the membrane structures as a set of pre-assembled vesicles within molecular simulation, it will be possible to determine numerically if the structural and stoichiometrical dynamics as described in the formalism are stable and represent the original design intention when implemented *in vitro*. By analysing the dynamic behaviour of the membrane structures described in P systems models, it will be possible to alter or constrain P Systems to create more realistic ALife models.

3.2 Dissipative Particle Dynamics

When attempting to model systems at the molecular scale, increasing the level of detail quickly leads to equations that are not solvable using analytical methods, and so it becomes necessary to perform numerical simulations to determine the dynamics of the system. Molecular Dynamics has become the standard tool for simulation of molecules in disciplines where quantitative accuracy is paramount, such as computational chemistry. However, the simulation length and time scales which are feasible in Molecular Dynamics with current hardware is limited by the complex and long range forces acting between particles, and so we turn instead to a more coarse grained model intended for the study of fluid dynam-

ics, Dissipative Particle Dynamics (DPD)[13]. Although the DPD model was originally proposed for the study of computational fluid dynamics, reformulations with respect to statistical mechanics[6], the relative ease with which inter-particle parameters can be derived[11, 10] and the inclusion of polymers within the model has meant that the method is used for simulation and modelling of a number of different equilibrium processes.

DPD employs an approach in which a certain number of molecules of a given type compose a single “bead” or coarse grained particle within the simulation, in the case of water each bead is equivalent to approximately three water molecules in volume. The total number of particles within the system is therefore greatly reduced when compared with the total number of molecules in the actual system. As the particles being simulated represent several molecules, the interactions between them must also be coarsened, and so the dynamics of the systems are described by a simple repulsive force (the conservative force), that declines linearly with inter-particle distance and conserves momentum. Despite the simplicity of the potentials the model has been shown to be very useful for describing hydrophobic interactions and has been successfully applied to the simulation of micelles, bilayer and vesicle formation [20, 10, 9]. One drawback with the DPD method is the incorrect reproduction of some transport properties, for example the schmidt number of fluids in DPD is much lower than experimentally observed values for real fluids[11]. However, as we seek a qualitative rather than quantitative understanding of vesicle based P-Systems, we consider this inaccuracy acceptable.

Three forces describe the pairwise, symmetric interactions between particles that are within the force interaction radius in DPD, these are the conservative force, as described above, and defined as follows:

$$F_{ij}^C = \alpha(1 - \frac{|r_{ij}|}{r_c}) \quad (2)$$

Where r_{ij} is the vector pointing from particle j to particle i , $|r_{ij}|$ is the distance between particle i and particle j , α is the maximum repulsion between the two particle types and r_c is the cut-off distance. The dissipative and random forces act together to define the system temperature, the dissipative force is defined as follows:

$$F_{ij}^D = -\gamma w^D(r_{ij})(\hat{r}_{ij} \cdot v_{ij})\hat{r}_{ij} \quad (3)$$

Where $\hat{r}_{ij} = \frac{r_{ij}}{|r_{ij}|}$, the unit vector pointing from particle j to particle i , and v_{ij} is the difference in velocity between particle j and particle i . The dissipative force acts to reduce the amount of energy in the system by reducing particle velocities if they are approaching one another.

The random force introduces energy into the system by inducing a force with random magnitude between particle pairs, and is shown below:

$$F_{ij}^R = \sigma w^R(r_{ij})\theta_{ij}\Delta t^{-1/2}\hat{r}_{ij} \quad (4)$$

Where θ_{ij} is a random variable with unit variance, zero mean, and no memory between time-steps or particle pairs.

In order to correctly reproduce the canonical temperature, the dissipative and random forces are coupled with weighting functions $w^D(r_{ij})$ for the dissipative force and $w^R(r_{ij})$ for the random force. One of these functions may be chosen arbitrarily, and we use the random force weighting proposed

by Groot and Warren[11]:

$$W^R(r) = \begin{cases} (1-r) & \text{when } (r < 1) \\ 0 & \text{otherwise} \end{cases} \quad (5)$$

The dissipative weight function is then derived from the following relation:

$$W^D(r) = [W^R(r)]^2 \quad (6)$$

The γ and σ parameters are used to set the system temperature and related as follows:

$$\sigma^2 = 2\gamma k_B T \quad (7)$$

where γ controls the magnitude of the dissipative force, and σ controls the magnitude of the random force. The total force acting between each pair of particles is then given by the summation of these three forces.

The equations of motion are integrated using a modified Velocity-Verlet method first described by Groot and Warren[11]. In order to differentiate between substances, the α parameter in the conservative force is set differently for different particle types, for example, interactions between amphiphile tail beads and water will be highly repulsive, and so have a high α value. Polymers are modelled in DPD by tying beads together with a simple harmonic spring force

$$F_{ij}^S = k(r_{ij} - r_0) \quad (8)$$

where k is the bond strength parameter, and r_0 is the preferred bond length. Constraints on bond angles can be included with a harmonic 3-body potential[14]

$$U_\theta = \frac{1}{2}K_\theta(\theta - \theta_0)^2 \quad (9)$$

Where K_θ is the angle force strength parameter, θ is the angle between the two bonds and θ_0 is the preferred angle.

We use a self developed parallel implementation of DPD designed to run on a distributed memory cluster. A simulation of 375,000 particles running for 10,000 DPD time units takes approximately 20 hours on 8 processors.

4. REALISING P SYSTEMS IN DPD

In our methodology P systems membranes delimiting regions are represented by vesicles composed of model amphiphiles. Biological vesicles are small volumes of fluid encapsulated by a phospholipid bilayer separating the internal volume from the external environment. They are formed and utilised within cells for transportation, metabolism and storage. While cell membranes are highly complex, functionalised structures performing a range of sensing and filtration tasks for the cell, P system membranes make no explicit reference to this functionality. Therefore we start with vesicle membranes, which can be considered as model cell membranes and add functionality as necessary with respect to the systems being modelled. Simulations of systems based on small vesicles (with diameters in the nanometre range) have already been successfully simulated in DPD[20].

Vesicle formation is a self-assembly process which occurs due to the hydrophobic exclusion of hydrocarbon chains in the amphiphiles. These chains disrupt the matrix of hydrogen bonds between the water molecules leading to unfavourable energy configurations [27]. In DPD, the hydrophobic effect can be modelled by correct parameterisation of a simple repulsive force acting between the solvent and hydrocarbon chain particles.

4.1 Amphiphile Parameters

For all simulated amphiphiles in this work, we use a model amphiphile which is similar to that used in [26] for simulation of fusion of vesicles with a planar bilayer. The amphiphiles are subdivided into volumes of 90\AA and the resulting beads are “bonded” using simple Hookean spring forces. The amphiphile configuration is shown in figure 1 where the red (darker) particles make up the hydrophilic head, and the green (lighter) particles compose the hydrophobic tails. All bonds between amphiphile particles have a preferred length of 0.7 DPD units² and a strength of $100k_bT$, an angle force with strength $6k_bT$ is applied to each sequential triplet of beads in the tail chains, with a preferred angle of 180° and an angle force with strength $3k_bT$ and preferred angle of 90° is applied at the point where the two tail chains join the head group. It was found that amphiphiles with these parameters rapidly self-assemble into vesicles.

We use the following conservative force parameters, for interactions between particles of the same type $\alpha = 25$, for tail-water bead interactions $\alpha = 80$, for tail-head interactions $\alpha = 80$ and for head-water interactions $\alpha = 15$. The σ and γ parameters are set to 3.00 and 4.5 respectively, resulting in a temperature equivalent to room temperature. The time step, Δt was set to 0.05 DPD time units and the empirical parameter for the Groot Warren integration scheme λ was set to 0.65 . To produce a corresponding representation of the P system within DPD, it is necessary to create an initial condition for the simulation containing the required structure of vesicles.

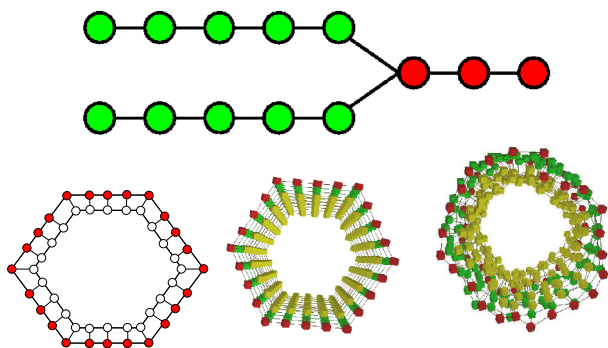


Figure 1: The structure of the amphiphile used (top), a schematic representation of a pore (left), the initial configuration (centre) and the configuration after 1000 time steps (right).

4.2 Modelling Pores

To facilitate the exchange of molecules between compartments we designed pores which assemble into the vesicle membrane. The model pore is a roughly circular inclusion and is constructed from two layers, the outer layer reproduces the hydrophobic/hydrophilic profile of the bilayer, and the inner layer is simply composed of tethered solvent particles, which act as a buffer between the hydrophobic beads in the outer layer and the solvent beads passing through the pore. Without this buffer, we found that amphiphiles would

²The unit length in DPD is the force interaction radius, as a single bead has a volume of 90\AA , the interaction radius is therefore 6.463\AA .

embed within the pore during the self-assembly process, effectively blocking it as water particles were unable to pass the hydrophobic tails of the amphiphile.

Figure 1 shows the pore design, the initial configuration of the pore, and the pore after simulation for 10,000 time steps; the darker particles in the external layer of the pore mimic the properties of the bilayer. All bond lengths between particles having a preferred length of 0.5 DPD units, and the maximum bond strength parameter k for all pore bonds was set to $100k_bT$. Although in the initial configuration the pores are hexagonal in shape, they quickly relax into a circular structure with a diameter of roughly 18\AA . Each pore has seven layers, the top and bottom layers contain hydrophilic particles in the outer layer, and the middle five layers contain hydrophobic particles. This profile was chosen to mimic the cross sectional dimensions of the bilayer.

5. MODELLING DIFFUSION

In order to demonstrate the multi-scale simulation methodology, we start with a very simple diffusive system containing two regions, one encapsulated within the other, and study the effect of poration on the encapsulated membrane by tracking the movement of solvent particles from within the region to the external environment. In biological membranes, transport proteins are of great importance, essentially acting as the interface between the intra-cellular and extracellular environments. The selective permeability afforded by these

proteins enables the nutrients of cell metabolism to pass into the cell, even against a concentration gradient, and waste to pass out of the cell without leaving the intra-cellular volume vulnerable to contamination[16]. Membrane pores vary greatly in complexity and diameter, from passive Porin proteins which select based on molecular size to large, complex ATP activated proton pumps and gated ion channels which open or close based on presence or absence of an activator, such as a ligand binding and inducing a conformational change in the protein.

5.1 Self-Assembling Porated Vesicles

In order to create the porated vesicle representing the P system region, DPD simulations were performed starting with a number of amphiphiles and pore polymers distributed randomly about the simulation space of 50^3 in DPD units with a number density of 3 , resulting in simulations containing $375,000$ particles in total. The simulation space was initialised by randomly distributing DMPC amphiphiles, such that 19% of the total number of particles belongs to DMPC polymers. The percentage of particles belonging to pores was then varied in the range of $0.25 - 1.5\%$ in increments of 0.25% (Table 1 shows the results of these simulations).

Each simulation was integrated for $200,000$ time-steps (and vesicles were found to typically form after $100,000$ steps), the resulting porated vesicles were then extracted from the simulation output and used as the initial conditions for the diffusion simulations. Figure 2 shows a snapshot of the vesicle self-assembly process, initial configuration for a porated vesicle formation simulation, a snapshot of a vesicle containing 7 pores within the membrane and the corresponding P system.

In order to measure the rate of diffusion, the internal solvent particles of the vesicle were tagged *yellow* (the interaction parameters for these tagged particles remained the

Table 1: Simulations of vesicle formation, with membrane pore inclusions

Total number of pores	Pores included in vesicle membrane	Resulting vesicle diameter estimated from volume(nm^3)
2	1	15.41
5	2	10.94
8	4	12.40
11	7	15.61
13	11	16.05
16	14	18.10

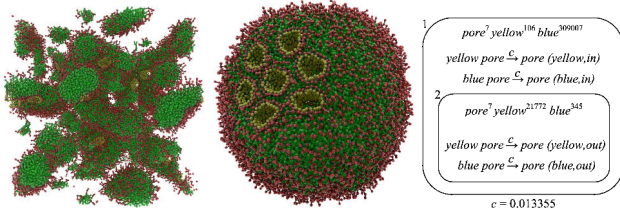


Figure 2: A snapshot of the DPD simulation showing the self-organisation of amphiphiles in micelles/vesicles at 20,000 time steps (left), a fully developed porated vesicle (at time step 200,000) (centre) and the P System mathematical structure related to the centre vesicle (right).

same as for normal solvent) and the vesicle was surrounded by un-tagged *blue* solvent. Table 2 shows number of tagged and un-tagged particles within each vesicle at the initial condition, and the length of each simulation in time steps.

Table 2: Initial particles counts within each vesicle

Number of pores	Number of tagged solvent particles	Number of un-tagged solvent particles	Simulation length (time-steps)
1	21120	173	2,400,000
2	7378	250	600,000
4	10900	169	400,000
7	21772	245	400,000
11	23265	767	400,000
14	32352	2182	400,000

5.2 A P System Model of Diffusion

We first attempt to model the above system with a stochastic P system and then improve it by correlating its behaviour with that of a DPD simulation, using the sets of parameters given in table 2 as the initial multisets for the P systems. The number of pores in the vesicle membrane is varied to investigate the effect each pore has on the rate of diffusion across the membrane.

The P system specification used in all experiments is:

- an alphabet of objects $O = \{yellow, blue, pore\}$;
- the membrane structure $\mu = [1 [2]_2]_1$, as shown in figure 2;
- 2 sets of communication rules R_m modelling diffusion

of yellow and blue particles in and out of region m :

$$R_1 = \begin{cases} yellow\ pore \xrightarrow{c} (pore, here) (yellow, in) \\ blue\ pore \xrightarrow{c} (pore, here) (blue, in) \end{cases}$$

$$R_2 = \begin{cases} yellow\ pore \xrightarrow{c} (pore, here) (yellow, out) \\ blue\ pore \xrightarrow{c} (pore, here) (blue, out) \end{cases}$$

In the initial experiment an arbitrary value of c of 0.001 was assigned to all rules, this ensured that the propensity of each rule was dependent solely on the relative concentration of yellow and blue particles within each membrane.

The initial multisets of this experiment were taken from the 1 pore DPD simulation noted in table 2. The nature of the simple rules employed here require a pore to be in both the internal and external regions as objects of one region are not visible to rules of another.

$$w_1 = yellow^{146} blue^{21149} pore^1$$

$$w_2 = yellow^{21120} blue^{173} pore^1$$

Figure 3 shows the movement of solvent objects between regions in the system over the simulation length. Yellow particles moved out of the vesicle and blue particles enter the vesicle due to high concentrations of these particles in regions 2 and 1 respectively. The simulation reached equilibrium after approximately 1,500,000 time steps. As the concentrations approached equality all rules have similar propensities and an equilibrium is maintained of approximately 10,000 yellow and 10,000 blue objects in each multiset. The simulation was terminated at 2,400,000 time steps.

In the equivalent DPD simulation, yellow tagged solvent exits while blue solvent enters until the levels of internal blue and internal yellow reach similar amounts, which is an equilibrium between internal and external concentrations of yellow. The disparity between the results of these two methodologies was reasoned to be due to the fairly constant size of the vesicle constraining the amount of solvent it can hold whereas a P system membrane is an abstract container and its multiset can take any integer number of objects. In this P system any configuration of initial multisets or membranes would eventually share the objects evenly.

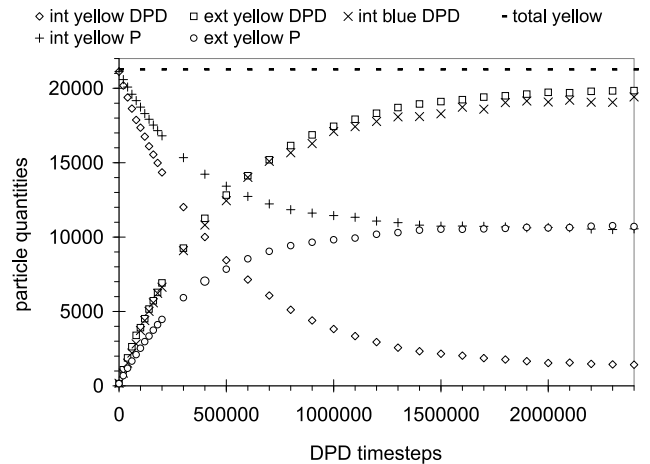


Figure 3: Equilibration in the initial P system model does not match DPD.

5.3 A Refined P Systems Model

Molar concentration is the number of moles per litre of volume. We refined the P system model above by introducing the notion of volume into the membrane definition and incorporating that volume into the propensity function. By definition solvent particles comprise the medium of the DPD simulation and as such are a suitable indication of volume within the system. The volume of a region is, in this instance, calculated to be the sum of the multiplicities of the yellow and blue particles in the initial multiset divided by 3, the number density in DPD. In systems with greater numbers of object types the cardinality of the multiset may be a more suitable alternative. In the improved model the propensity function was changed to divide the rate constant by the volume:

$$\text{propensity} = \frac{c}{V} |water||pore|$$

In this way volume acts to reduce the propensity of rules in larger volumes, modelling the reduced likelihood of a particular particle being in the proximity of a pore.

By simulating the refined P system with initial multisets shown in table 2 and comparing the timecourses a rate constant of 0.013355 was empirically determined.

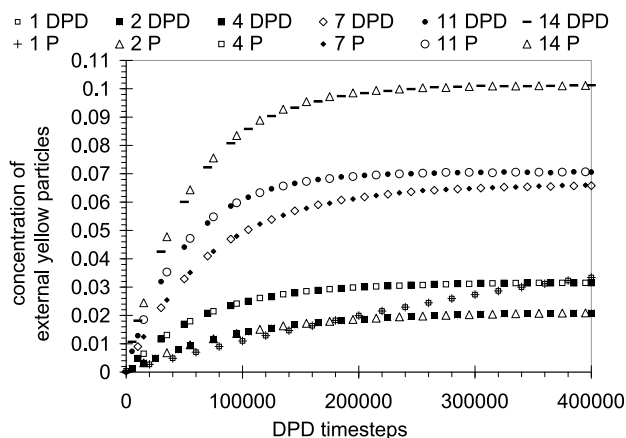


Figure 4: Plot of the concentration of external yellow particles from both P system and the DPD simulations over 400,000 time steps which a range of pores.

With this value for c the behaviour of the P system very closely matches that of the DPD simulations as shown in Figure 4. It can be observed that simulations with more pores had a greater initial rate of yellow efflux and achieved greater concentrations. The exception is the single simulation which achieves a higher final concentration than the simulation with 2 pores. This is due firstly to the number of pores in the membrane multiplying the propensity of the rules, and secondly the size of vesicles required to contain more pores which determined the numbers of tagged particles in the system.

Figure 5 plots of the change in concentration over time during the first 50,000 steps against the number of pores in the simulation. The rate of efflux for yellow particles increases linearly with the number of pores in the membrane, increasing by 0.0037 for each additional pore.

The results show that despite the difference in scale, the

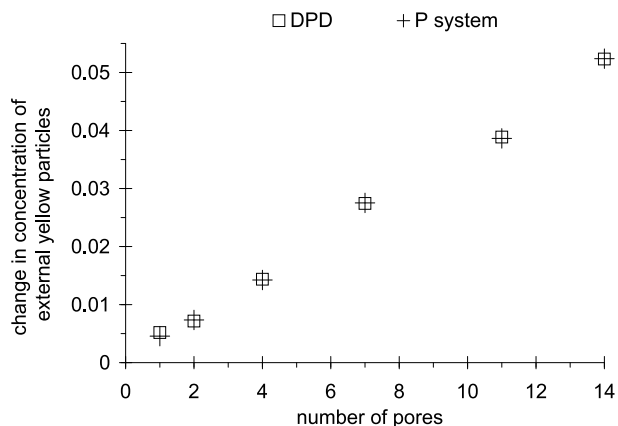


Figure 5: Rate of yellow efflux is proportional to the number of pores in a membrane.

P systems model can be calibrated to match the observed behaviour of diffusion in the DPD model. By linking the two in this way, we bridge the scale gap between the abstract model and more realistic low level simulation.

6. CONCLUSIONS

In this paper we have outlined a prototype multi-scale model for simulation and design of artificial life processes at length scales approaching those of the smallest bacteria. By coupling the P systems formalism with a coarse grained particle dynamics simulation, and calibrating the high level model with results from the low level simulation we have shown that designs can be expressed quickly as P systems, and high level stochastic analysis performed. If the design shows promise, then the system can be represented within DPD as a set of vesicles, and the dynamics of the system fully explored. We show that this method is viable by analysing diffusion in a simple system of porated membranes, and by refining the model described we produce a stochastic P system that correlates well with the behaviour from the DPD simulation. This P system can then be used in more complex designs of artificial life systems, or for gaining insight into the behaviour of the system over longer time scales. Notably, once a P system model has been fine tuned to a DPD simulation it is possible to perform model checking, we will report on this in a future publication.

We have produced a P systems model that is capable of modelling diffusion in DPD by parameterisation with a lower level model. Although the system being modelled is very simplistic when compared to ALife processes, the results illustrate a proof of concept for the modelling framework. An extension of P systems that includes volume as a requisite and mutable property of abstract membranes that model biological membranes could be used to modelling cell growth and division.

As computational power increases, it will become possible to explore in simulation larger more complex systems in increasing levels of details, and so we see this as a first step towards a multiscale simulation environment allowing design and evaluation. The P Systems and DPD models are chosen as they are relevant in the simulation of artificial life. However, there is no reason why DPD could not be re-

placed by Molecular Dynamics when the computing power for simulations at the same scale as DPD become available.

In future work, we intend to explore more complex systems and ALife processes with the modelling framework, with the addition of chemical reactions to the DPD model, so that P system rules can be more faithfully represented. This could be done, among others by a judicious incorporation of the GARD modelling tools, which are based on explicit incorporation of a set of chemical reactions, including membrane entry and exit of molecules from the environment, as well as metabolic interconversions.

7. ACKNOWLEDGEMENTS

This work was supported by EPSRC grants EP/017215/1 and EP/D021847/1, and BBSRC grant BB/F01855X/1.

8. REFERENCES

- [1] F. Bernardini, M. Gheorghe, and N. Krasnogor. Quorum sensing P systems. *Theoretical Computer Science*, 371(1-2):20–33, February 2007.
- [2] L. Bianco, D. Pescini, P. Siepmann, N. Krasnogor, F. Romero-Campero, and M. Gheorghe. Towards a P systems pseudomonas quorum sensing model. *Lecture Notes in Computer Science*, Volume 4361/2006, pages 197–214. Springer, 2006.
- [3] A. G. Cairns-Smith. The origin of life and the nature of the primitive gene. *Journal of Theoretical Biology*, 10(1):53–88, Jan. 1966.
- [4] L. Cronin, N. Krasnogor, B. G. Davis, C. Alexander, N. Robertson, J. H. G. Steinke, S. L. M. Schroeder, A. N. Khlobystov, G. Cooper, P. M. Gardner, P. Siepmann, B. J. Whitaker, and D. Marsh. The imitation game - a computational chemical approach to recognizing life. *Nat Biotech*, 24(10):1203–1206, Oct. 2006.
- [5] P. Eggenberger. Evolving morphologies of simulated 3d organisms based on differential gene expression, 1997.
- [6] P. Español and P. Warren. Statistical-mechanics of dissipative particle dynamics. *Europhys. Lett.*, 30:191–196, 1995.
- [7] N. Geard and J. Wiles. A gene network model for developing cell lineages. *Artif. Life*, 11(3):249–268, 2005.
- [8] D. T. Gillespie. Stochastic simulation of chemical kinetics. *Annu. Rev. Phys. Chem.*, 58:35–55, 2007.
- [9] A. Grafmuller, J. Shillcock, and R. Lipowsky. Pathway of membrane fusion with two tension-dependent energy barriers. *Phys. Rev. Lett.*, 98(21):218101, 2007.
- [10] R. D. Groot and K. L. Rabone. Mesoscopic simulation of cell membrane damage, morphology change and rupture by nonionic surfactants. *Biophys. Journal*, 81:725–736, 2001.
- [11] R. D. Groot and P. Warren. Dissipative particle dynamics: Bridging the gap between atomistic and mesoscopic simulation. *J. Chem. Phys.*, 107:4423–4435, Sept. 1997.
- [12] D. Harel. A turing-like test for biological modeling. *Nat Biotech*, 23(4):495–496, Apr. 2005.
- [13] P. Hoogerbrugge and J. Koelman. Simulating microscopic hydrodynamic phenomena with dissipative particle dynamics. *Europhysics Letters*, 19:155, June 1992.
- [14] M. Kranenburg, M. Venturoli, and B. Smit. Phase behavior and induced interdigitation in bilayers studied with dissipative particle dynamics. *J. Phys. Chem. B*, 107(41):11491–11501, 2003.
- [15] C. Lattaud and S. Bornhofen. Advances in artificial life: A brief history of evolutionary individual based models. In *7th International Conference on Artificial Evolution*, pages 226–237. Springer, 2005.
- [16] H. Lodish, A. Berk, C. Kaiser, M. Krieger, M. Scott, A. Bretscher, H. Ploegh, and P. Matsudaira. *Molecular Cell Biology*. W.H. Freeman & Co Ltd, 2007.
- [17] P. L. Luisi. Self-reproduction of micelles and vesicles: Models for the mechanisms of life from the perspective of compartmented chemistry. In S. A. R. I. Prigogine, editor, *Advances in Chemical Physics*, pages 425–438. John Wiley & Sons, Inc, 1996.
- [18] J. V. Neumann. *Theory of Self-Reproducing Automata*. University of Illinois Press, Champaign, IL, USA, 1966.
- [19] A. Oparin. *Origin Of Life On Earth*. Elsevier, 1959.
- [20] V. Ortiz, S. Nielsen, D. Discher, M. Klein, R. Lipowsky, and J. Shillcock. Dissipative particle dynamics simulations of polymersomes. *Journal of Physical Chemistry B*, 109(37):17708–17714, 2005.
- [21] G. Paun. *Membrane Computing: An Introduction*. Springer-Verlag New York, Inc., 2002.
- [22] A. Pohorille and D. Deamer. Artificial cells: prospects for biotechnology. *Trends in Biotechnology*, 20(3):123–128, Mar. 2002.
- [23] S. Rasmussen, L. Chen, M. Nilsson, and S. Abe. Bridging nonliving and living matter. *Artif. Life*, 9(3):269–316, 2003.
- [24] C. W. Reynolds. Flocks, herds and schools: A distributed behavioral model. *SIGGRAPH Comput. Graph.*, 21(4):25–34, 1987.
- [25] D. Segré, D. Ben-Eli, D. W. Deamer, and D. Lancet. The lipid world. *Origins of Life and Evolution of Biospheres*, 31(1):119–145, Feb. 2001.
- [26] J. Shillcock and R. Lipowsky. Tension-induced fusion of bilayer membranes and vesicles. *Nature Materials*, 4:225–228, 2005.
- [27] C. Tanford. *The Hydrophobic Effect*. New York: Wiley, 1973.
- [28] D. Segré, D. Ben-Eli and D. Lancet. Compositional genomes: prebiotic information transfer in mutually catalytic noncovalent assemblies. *Proc. Natl. Acad. Sci. USA* (2000), 97(8), 4112–4117.
- [29] B. Shenhav, D. Segré and D. Lancet. Mesobiotic emergence: Molecular and ensemble complexity in early evolution *Advances in Complex Systems* (2003), 6(1), 15–35.
- [30] B. Shenhav, A. Solomon, D. Lancet and R. Kafri. Early Systems Biology and Prebiotic Networks *C. Transactions on Computational Systems Biology*, LNCS 3380, p. 14–17, 2005.
- [31] A. Hunding, F. Kepes, D. Lancet, A. Minsky, V. Norris, D. Raine, K. Sriram, R. Root-Bernstein. Compositional Complementarity and Prebiotic Ecology in the Origin of Life. *Bioessays*. 2006 Apr;28(4):399–412