Gene Regulation in a Particle Metabolome

Simon Hickinbotham, Edward Clark, Susan Stepney, Tim Clarke and Peter Young

Abstract— **The bacterial genome is well understood by biologists. Although its efficiency and adaptability should make it a good model for evolutionary algorithms, the bacterial genome is tightly coupled with the components of the bacterial metabolism, referred to here as the *metabolome*. This paper explores an approach to modelling an artificial bacterial metabolome in an efficient and modular manner, so that analogues of bacterial genome organisation and gene regulation can be implemented in evolutionary algorithms. We propose a particulate model of bacterial metabolic pathways in which the constituents drift in a fixed, limited space and obey a limited set of biologically plausible reaction rules. The potential of this model is demonstrated by creating a network that is capable of appropriate behavioural switching that can be observed in bacteria.**

I. Introduction

We observe from the bio-diversity present in nature the power of biochemical evolution to *build*. This property is not so clearly demonstrated in artificial evolutionary algorithms (EAs), which tend to serve more as optimisers [1]. Current attempts to improve the ability of EAs to create functional structures take one of three routes: modifying canonical EA mechanisms via (for example) new configurations of crossover or mutation [2], augmenting the system with other computational devices such as neural networks [3], or taking further inspiration from biology [4]. Our ambition is to take the biological route, but to take a few steps back from current artificial EA systems and draw inspiration directly from so-called "simple" unicellular life forms, specifically the prokaryotes (bacteria) and their postulated precursors in the evolution of the early earth. Central to this approach is the idea that the metabolism acts as a sorting-house for signals from the environment and from the genome. The metabolism is attractive for several reasons: it reflects the response of the genome to the current state of the environment on a range of time scales; it is able to deal with information in a variety of forms; it is capable of switching behaviour as a result of change in stimulus. These desirable properties are all coded for in the bacterial genome.

Given that bacteria demonstrate that a genome can be used to build information-processing "factories", why have they not been used more extensively as templates for evolutionary algorithms? The answer is that the bacterial metabolism is a highly complex network of interacting three dimensional structures which biologists have been working on modelling and understanding for the past 150 years [5]. These works all have the goal of understanding elements of the prokaryote metabolism. Our goal is different: developing a model metabolism that is sufficiently rich to allow useful experiments with regulation of gene expression within EAs to be carried out. We find the term *metabolome* useful here. The metabolome is simply the set of (small) molecules that make up a biological system along with reactions that occur between them, whereas the metoblism defines the "physics" of the system, which is not encoded on the genome.

We require an appropriate abstraction of bacterial metabolism that preserves their complexity and robustness but which can also be encoded in some artificial genetic representation such that evolutionary experiments can be conducted. We want to see what computational features and problems exist at the metabolic level, so that we don't waste time constructing genetic systems that do not encode them with appropriate detail. In this sense, the model is "top down". But we want to emphasise that we are aiming for a "pluggable" model, in which the representation of the metabolic processing unit is separated from the genomic and protein/enzyme/molecular representations. Different applications of this model will require different resolutions in these three domains. We do not therefore concern ourselves with specific issues of three dimensional shape of metabolites, with all the implications for protein folding and binding that follow. Nor do we want to model our metabolism as a continuous distribution of concentrations of solvents, since that removes the possibility for local variation of individual metabolites that is a necessary part of evolution. Finally, we are not concerned about faithfully simulating biological reaction rates and metabolite counts of bacterial metabolomes, since we recognise that the computational burden would probably be too heavy even for that.

This paper describes our particle metabolome model, and demonstrates how it can be used to engineer self-regulatory control in a virtual organism. As a test of the versatility of the resulting metabolome, we describe how it can be used to model gene regulatory control of the enzyme complement of an artificial metabolism. We take inspiration from diauxy, the regulation of the metabolism of lactose, which is an alternative and less energetically beneficial dietary substrate to glucose [6]. Note that our goal is to demonstrate that this *type* of control can be implemented in the system we describe.

II. The particle metabolism

We are constrained by the idea that metabolome particles can not make reference to some cell-level instruction set that determines what metabolic reactions are permissible, since our long term goal is to evolve metabolites that are

This work is part of the Plazzmid project, funded by EPSRC grant EP/F031033/1. All authors are affiliated with the York Centre for Complex Systems Analysis, University of York, Heslington, York YO1 5DD, UK. Simon Hickinbotham; email: sjh@cs.york.ac.uk), Ed Clark and Susan Stepney are at the Department of Computer Science, Tim Clarke is at the Department of Electronics and Peter Young is at the Department of Biology.

"aware" of only their own reaction rules, without reference to some global controller. We use simple reactive computational entities (agents) to represent these metabolites. This means that the agents that we build cannot be represented as concentrations, and must exist as particles, with a location in space and a localised zone of interaction. Moreover, continuous representations of very low concentrations of particles would not model the stochasticity of the system very well. In order to minimise computational overheads the particles must exist in small numbers (see table II) compared to the number of enzymes in a single bacterium.

There are several approaches to particulate representation of chemical systems. Among the most well-known are Gillespie's Stochastic Simulation Algorithm (SSA) [7], and the faster extension of it by Gibson and Bruck [8], in both of which no spatial component is realised. These are efficient approaches but they demand that the system is well-mixed which is difficult to guarantee in all but the simplest systems. Green's function reaction dynamics [9] can be used to specify a spatial representation of a system but like both SSAs, the dynamics do not feature a uniform time step, making it difficult to devise an interface with an external signalling component. Stochsim [10] is an alternative that is most similar to our approach but it has not been developed *specifically* for evolutionary experiments. Our model exploits the freedom from the pressure to simulate biological quantities and rates, and seeks a way to explore and simulate the evolutionary mechanisms observed in biological systems.

This particle-based specification of the metabolome is also reminiscent of Couzin et al's model of spatial organisation of animal groups [11], in which individuals are modelled as particles that have predetermined radii associated with them. Each "action radius" carries with it a specification of the events that occur should another individual appear inside that radius.

Our agents are simpler than higher organisms, and require only a single action radius within which a check for neighbours is made. Energy is required for reactions to take place, and some reactions also yield energy. Energy is modelled as a continuum since the computational cost of representing energy via individual particles is too great. This is an appropriate abstraction for the problem domain, and is in line with current practice in systems biology [12]. Waste materials, and materials required for construction of agents are assumed to be approximately constant and are not represented in the current model.

A. Time, space and motion

We use a discrete time interval to control the rates of reaction and motion. In an elemental time step t_i, a set of agents will be present in the metabolome.

The model is currently implemented in two dimensions, which is a compromise between the need for some notion of space in the model and computational expediency.

Since speed is defined as distance per unit time, the effect of the size of the arena is linked to the time unit via the motion of the agents. Motion of a fixed distance in a random direction occurs at every time step for each agent. At a conceptual level, the metabolism of the 'cell' is contained within a simple membrane. This is encoded via a change in random motion should an agent come within a specified radius of the perimeter - motion at the next step is always towards the centre of the cell. Motion of a particle into the cell is illustrated in figure 1.

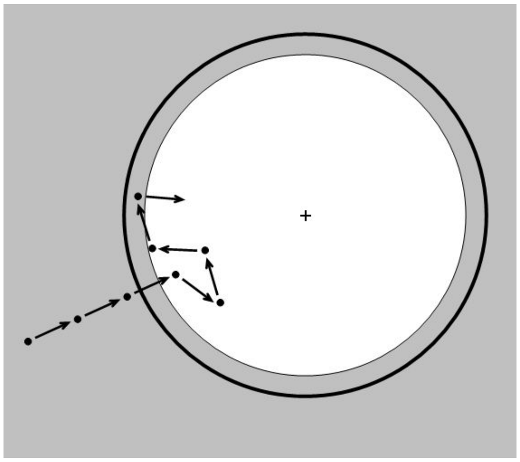

Fig. 1. Influx and motion in a particle cell. Particles outside the cell (grey region) move towards the centre. Once inside the cell, motion proceeds in a random direction at each time step. Particles cannot escape the cell radius.

With our environment defined, we can illustrate how the algorithm proceeds by reference to the following pseudocode:

```
while(alive)
    reactions()
    motion()
    influx()
    if(can_divide())
        divide()
    if(has_died())
        alive := FALSE
```

In each unit of discrete time, the reactions in the system are carried out first. This is when the constituents of the next time step are generated. The constituents of the next time step are then subjected to motion, and any influx of new agents occurs. Having finished setting up the next time step, two tests are made. We test whether the cell is healthy enough to "divide", and then we test whether the cell has "died". These events are described below.

B. Reactions

Ideally, we would like a system in which any possible reaction between any number of constituents can occur. It would be a daunting task to encode all possibilities within a single time step. Fortunately the biological reality shows that complex reactions in major metabolic pathways tend to occur as a sequence of simpler reactions spread through time [13]. In our model, we limit the constituents of any individual reaction in a time step to two at most. We consider that there are only three types of reaction for an agent in a single time step. permissible rules take the following form:

Binding:	$\mathsf{A} + \mathsf{B} \to \mathsf{C}$
Dissociation:	$\mathsf{A} \to \mathsf{B} + \mathsf{C}$
Decay:	$\mathsf{A} \to \emptyset$

where A, B and C are agents, and Ø represents the absence of any product. Influx of new agents to the metabolome is a special case, and can be specified by randomly placing the required number of new agents at the perimeter of the cell at specified time periods.

At each time step, an agent is randomly selected from the metabolome, and tests are made to determine whether a reaction takes place, and what the reaction should be. The agents resulting from every reaction are placed in the subsequent time step. This process is repeated until there are no agents remaining in the current time step. In pseudocode:

```
A := random_agent()
B := closest_reactant(A)
if(B is not NULL)
  bind(A,B)
else
  dissoc_or_decay(A)
```

1) Binding: These rules take priority, since they require two agents to be sufficiently close for the reaction to be tested. At each time step, an agent should query its immediate environment, and react with objects that are sufficiently close for a reaction to be permissible.

A single binding reaction is processed as follows. An agent A is selected at random. All other particles in the system are checked against A, with the goal of finding an agent B that is within a fixed radius of A and for which there is a reaction rule specifying that A and B can bind. Note that it is more computationally efficient to hold reactions in a global rule table, which we have done here. Our long term aim is for each agent to have an internal specification of the reactions it can be involved in, to accommodate evolutionary change. The binding reaction has a rate attached to it. The reaction rate is modelled stochastically - a uniformly distributed random number between 0 and 1 is selected, and if it is lower than the reaction rate, then the reaction proceeds - this is analogous to the probability of binding between metabolites in a biological system.

The products of the reaction (zero, one or two agents) are placed in the set for the subsequent time step t_{i+1}. In this way, the set of agents for t_i undergo changes which form the agent set in t_{i+1}. Note that if the reaction does not occur, then the candidate agent is simply moved from t_i to t_{i+1}.

2) Dissociation and Decay: Binding rules cannot fire if pairs of agents are not sufficiently close. Where this is the case, the remaining reactions are tested. Since dissociation can happen only to compound agents, and decay can happen only to atomic agents, sensible configurations of a network ensure there is no clash of these rules.

Dissociation and decay have rates and energy costs associated with them in a similar manner to binding. Where a compound agent is in equilibrium with two constituent agents, the sum of binding and dissociation rates should equal one, and have zero or minimal cost. Decay rates are usually very slow for enzymes. In our model, energy is produced by the decay of the final product of catalysis.

Motion is an issue for dissociation, since there is a risk that the products of dissociation may immediately associate again if an appropriate rule is specified. For a dissociation rule $\mathsf{A} \to \mathsf{B} + \mathsf{C}$, the first product of the dissociation rule B assumes the position of the original compound agent A. The second product agent C is then moved in a random direction to 1.1 times the reaction radius of the first product agent. Both agents will be moved again in t_{i+1}, giving a small chance of being in the binding radius in the next time step.

The process of decay is essential to the model, since if enzymes didn't decay it would be much more difficult to control the constituents in the metabolome. Thus all particles except genes are transient - substrates are converted to energy, enzymes decay, and bound complexes dissociate to constituents. We can calculate the half life h of an enzyme using:

$$h = \frac{\log(1/2)}{\log(1-r)} \tag{1}$$

where r is the decay constant for the molecule.

3) Energy and Raw Materials: We currently model energy as a continuum, which is supplemented by the products of successful metabolism of the substrates that enter the cell. Energy is expended in large quantities during gene expression reactions and smaller quantities during catalytic reactions. Reactions cannot proceed if there is not sufficient energy available. If there is no energy available, the cell dies. We select agents in a random sequence when processing a particular time step. Agents at the back of the queue might find that there may not be energy available for their reactions in the time step. Note though that we are not constrained by physics, and can build any energetic cost or benefit into reactions as we choose.

III. Components of a model metabolome

Having specified the environment within which our virtual cells exist, we now describe the components of the metabolomes that we have been analysing experimentally. There are three chemical mechanisms forming a network from this metabolome: catalysis of model substrates S to make energy; gene expression of the enzymes E required for catalysis and regulation; and gene regulatory elements R. We use these elements to construct a metabolome that can exhibit dietary switching, inspired by respiration via glucose and lactose. Our motivation for doing so is that the regulation of the lactose metabolism is well understood, and widely used to demonstrate how gene regulation works. Respiration of glucose is energetically preferable, so bacteria manufacture enzymes for metabolising lactose only when

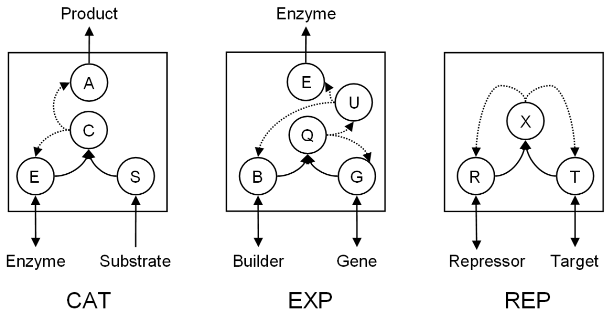

Fig. 2. Reaction complexes in an artificial metabolome. CAT = Catalysis, EXP = Gene Expression and REP = Repression. Binding reactions are shown as solid lines and Dissociation reactions are shown as dotted lines.

there is no glucose available *and* lactose is present. In the network presented here, we claim only a very loose analogy between glucose and S_1 and lactose and S_2.

Reaction	Rate	Cost
Catalysis		
$[S_1, S_2] + E \rightarrow C$	0.95	-2
$C \rightarrow E + [S_1, A]$	0.05	0
Decay		
$A \rightarrow e$	0.0025	20
$[E, B, R] \rightarrow *$	0.00004	0
Gene Expression		
$[G_P, G_1, G_2] + B \rightarrow Q$	0.1	-200
$G_R + B \rightarrow Q$	0.1	-0
$Q \rightarrow G_* + U$	0.001	-2
$U \rightarrow B + [B, E_*, R]$	0.01	-2
Regulation		
$G_2 + R \rightarrow X_{G_2,R}$	0.9995	0
$X_{G_2,R} \rightarrow G_2 + R$	0.0005	0
$S_2 + R \rightarrow X_{S_2,R}$	0.999	0
$X_{S_2,R} \rightarrow S_2 + R$	0.001	0
$G_2 + S_1 \rightarrow X_{G_2,S_1}$	0.99	0
$X_{G_2,S_1} \rightarrow G_2 + S_1$	0.01	0

TABLE I

REACTION RATES FOR A GENE REGULATORY NETWORK. LISTS INDICATE ALTERNATIVE REACTANTS

A. Reaction Complexes

Figure 2 shows how catalysis, gene expression, and regulation are achieved in a network built using the reaction rules specified in section II-B. The boxes encapsulate the chains of reactions that achieve an analogue of the biological process when appropriate pairs of reactants meet, and are useful in that they eliminate the need to represent transient intermediate complexes in detailed network diagrams. The individual reaction rules used to build these sub-networks are shown in table I. Catalysis, CAT is modelled via a two-step process - the binding together of substrate and enzyme to form an intermediate complex C, followed by the dissociation of the complex to the enzyme and the product. Gene expression EXP is rather more complex than catalysis. Our model has a loose analogy with RNA world models, in that genetic material resides in free-floating particles which we call genes G [14]. There is usually more than one copy of a gene for a particular enzyme in the metabolism. In addition to genes, we have an “enzyme builder” B, loosely analogous to a ribosome, that is assigned the task of constructing the enzyme from the gene. B and G associate to form a series of intermediate complexes Q and U, with the net result that the count of B and G in the metabolism is preserved whilst a new enzyme particle has been created. There is no direct analogy in biology for this mechanism, but it is a computationally efficient means of achieving gene expression within our permitted reaction rules. Note also that B has to be available to build copies of itself. Repression REP is a simple binding and dissociation pair. The networks presented here require regulation via a repressor. The substrate and repressor R bind to form a complex X that is metabolically inert. Note though that it is just as straightforward to construct networks in which X is an active component rather than a deactivated one.

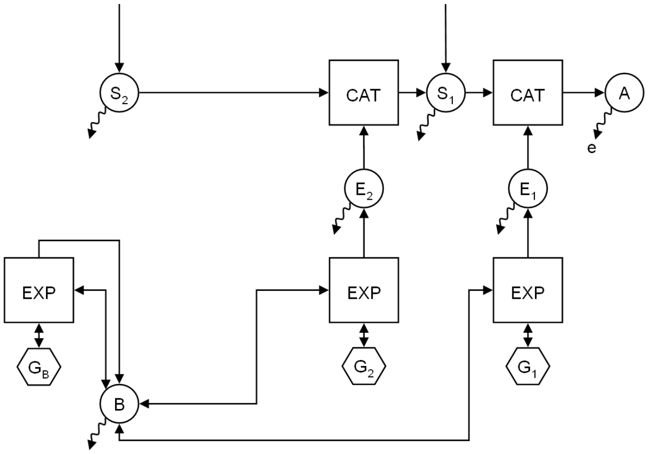

Fig. 3. An unregulated diauxic network. Circles indicate metabolites. Boxes indicate metabolic reactions as described in figure 2. Hexagons indicate genes. Straight lines indicate reaction pathways. Wavy lines indicate decay. G_2 is expressed regardless of whether there is any S_2 available to metabolise, or any need to metabolise S_2 in the absence of S_1.

B. A metabolic network for diauxy

An unregulated diauxic network is shown in figure 3. Substrates enter the system and undergo catalysis to produce the cell’s energy source A (top row). A releases energy into the system on decay. Gene expression pathways for the two enzymes E_1 and E_2 and the builder B are shown at the bottom of the figure. B associates with genes G_B, G_2 and G_1 to build the enzymes needed to run the network.

Our network model can be extended to regulate gene expression as shown in figure 4. The repression pathways are shown in the middle row of the figure. This requires the addition of a repressor molecule R which is expressed

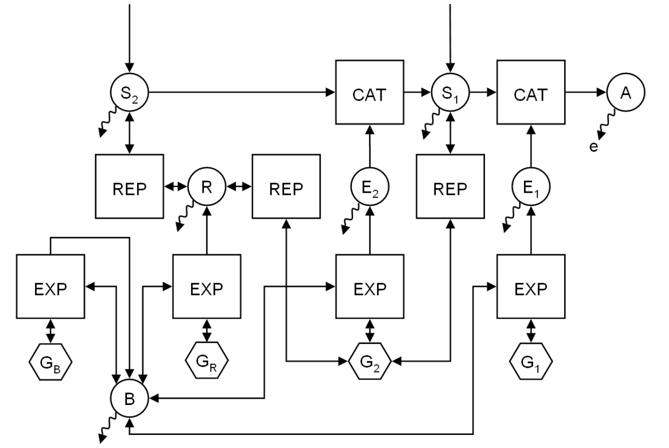

Fig. 4. A regulated diauxic network. Symbols are the same as for figure 3. Presence of S_2 and absence of S_1 are required for expression of the G_2.

from the gene G_R. Regulation of E_2 expression is achieved at the genome level via two switches. Positive regulation occurs via R which binds both to the gene for the E_2 and the substrate S_2. Because R is constantly associating and dissociating with these two particle types, S_2 mops up free R and thus makes the gene available for gene expression. Negative regulation is required to prevent E_2 being expressed when S_1 is available, since S_1 requires only one enzyme to metabolise it and is therefore energetically favourable to S_2. To facilitate this, we allow binding and dissociation of S_1 with G_2. Thus the gene G_2 is unavailable for gene expression in the presence of S_1, and no E_2 is produced. Note that this system exhibits a “leakiness” similar to that observed in biological systems since the repression complexes are constantly coming in and out of association. There is always the chance that B can bind with a repression target as it emerges from association from a repressor, albeit temporarily. We are never in the position to fully prevent a reaction occurring by offering an alternative pathway, and the interactions of sinks for a particular substrate is part of what gives biological systems such rich behaviour. Thus, the reaction rates are key in controlling the dynamics of gene regulation. We anticipate that this will be a key ingredient of the evolutionary experiments that this network will be used for.

C. Cell Cycle

It is possible that the metabolome could maintain particular quantities of metabolites indefinitely, although given the stochastic nature of our system, this is unlikely. If left to run for a sufficient length of time, a metabolome will either peter out to a point where there are no reactions possible, or accumulate ever larger quantities of metabolites. These situations have biological analogues - death or reproduction by cell division respectively. These phenomena are useful indicators of the performance of a metabolome. We detail here how these events are triggered.

Entity	Quantity
Enzymes	
$[P, R, E_1, E_2]$	6
Genes	
G_P	2
$[G_R, G_1, G_2]$	3
Energy	
e	1600

TABLE II

INITIAL QUANTITIES OF METABOLITES AND ENERGY IN A GENE REGULATORY NETWORK. LISTS INDICATE ALTERNATIVE REACTANTS.

1) Cell division: In addition to the substrate environment, we must also put in place some measure of growth of the cell in the system if we are to measure how successfully a particular metabolism responds to its environment. To do this, we simulate cell division in a very simple fashion. When the number of protein builders in a cell (whether free floating or in association) reaches a constant τ, the cell divides. In our trials for different substrate environments, we set $\tau = 20$. Rather than follow an ever increasing number of progeny from some seed, we follow a single lineage by retaining only one of the daughter cells. This process is achieved as follows. Cell division is organised spatially. When the conditions for division are met in a mother cell, all particles with a negative x coordinate are destroyed. The remaining contents are then redistributed randomly throughout the new daughter cell. In pseudocode:

```
for all agents A
  if(A->xcoord <0)
    destroy(A)
  else
    random_motion(A)
replenish_genes()
```

Since we are simulating cell division, it is necessary to ensure that the number of genes (indicated by hexagons in figure 4, with quantities described in table II) is preserved. We therefore survey the contents of the daughter cell to determine whether the full set of genes is present, and replenish them as necessary.

2) Death: A straightforward way to determine whether a cell has died is to check its energy budget. If there is no energy available, no reactions can proceed and so death can be pronounced on the cell. However, there are other situations where death is inevitable. For example, if the count of builders B in the cell goes to zero, then no enzymes E or builders can be constructed. Similarly, if any of the enzymes in the chain run out at a time when there is not sufficient energy available to replace them, then the cell will also die. We test for these situations at every time step, and do not proceed if any of these conditions are met.

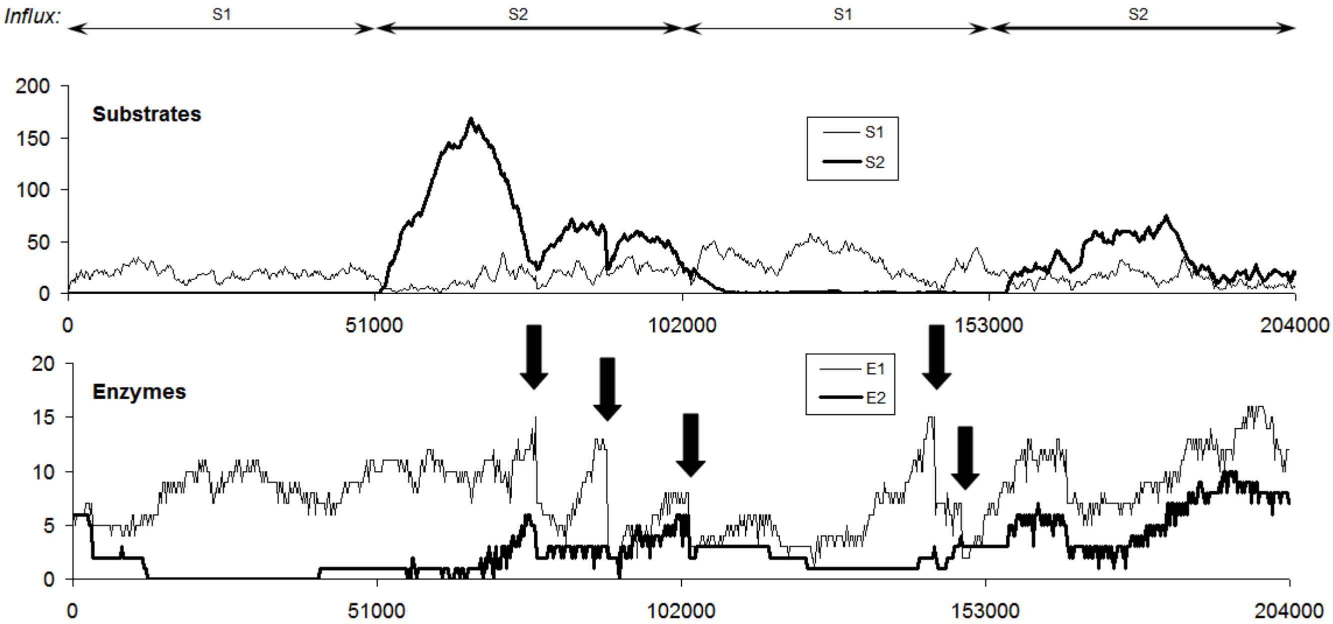

Fig. 5. Enzyme expression in response to substrate switching with $\tau = 20$. Levels of substrate in the cell are shown on the top graph, and levels of enzymes are shown on the bottom graph. Cell divisions are shown with black arrows. E_2 accumulates in the presence of S_2.

IV. Experimental evaluation

We have implemented the reaction network described above in C++, and present here a qualitative evaluation of the resulting network characteristics. Our goal is to demonstrate that the metabolome responds appropriately to an environmental fluctuation of substrates by regulating gene expression. Here we describe three experiments that we have conducted to evaluate the properties of our model.

A. Survival in a mixed substrate environment

In order to demonstrate the switching behaviour of our network, we have run a series of trials on the metabolome that it represents. In these trials, the virtual metabolism undergoes periods of immersion in one or other of the substrates S_1 and S_2 by setting stochastic influx rates limited to a maximum of 0.02 and 0.0225 units per time step respectively. The slightly higher value of influx of S_2 was chosen to compensate for the higher metabolic cost of processing it. This means that manufacture of E_2 is the only extra cost of metabolism of S_2. The metabolism must process these substrates in order to generate energy, which is subsequently used to build new enzymes to replace those that have decayed. As shown in table I, the enzymes in our system have a decay rate r of 0.00004. This value was selected so that the enzymes would have sufficient time to “earn their keep” by playing their part in the generation of sufficient energy to build more enzymes. This means each enzyme has a half-life of $h \approx 17,000$ time-steps. We consider that three half lives is sufficient time in one environment to demonstrate switching. We therefore switch between influx of S_1 and S_2 every 51,000 time steps. The reaction rates and initial quantities for the metabolome were selected by empirical trial, and are shown in table I and II respectively.

We have found that relative speed and costs of reactions is of critical importance to the functioning of the network. In particular, gene expression needs to be slower than catalytic and regulatory reactions, since otherwise regulation is swamped by the production of new enzymes. Gene expression is also the principal energy sink of the network. However the expression of regulatory molecules is much cheaper than that of catalytic enzymes, so that there is some benefit in expressing regulatory molecules over the enzymes they are supposed to regulate. To emphasise this and remove any issues regarding tuning this value, we have currently made the expression of the regulatory enzyme R free.

An example of part of a typical experimental trial is shown in figure 5. The trial commences with an S_1 diet. The six E_2 enzymes that the trial commences with decay away quickly, and are not replenished by expression of G_2 since G_2 is occupied in repression complexes with both S_1 and R. By contrast, G_1 is available to bind with B and manufacture E_1, and so a cycle is set up of metabolism of S_1 by E_1 to create energy, which is then used to replenish E_1 via expression of G_1. Note that E_1 must be constantly present to metabolise the product from E_2 into available energy. During S_2 phases, the plot line for E_2 is more jagged, since E_2 is binding with S_2 to carry out catalysis.

The diet switches to S_2 at $t = 51,000$. We see a build-up of S_2 in the cell, as there is no machinery available to process it. In time, the genes dissociate out of the repression complexes that have prevented gene expression and begin to express E_2. The cell processes the food surplus, and the available energy is then used to fund three divisions before

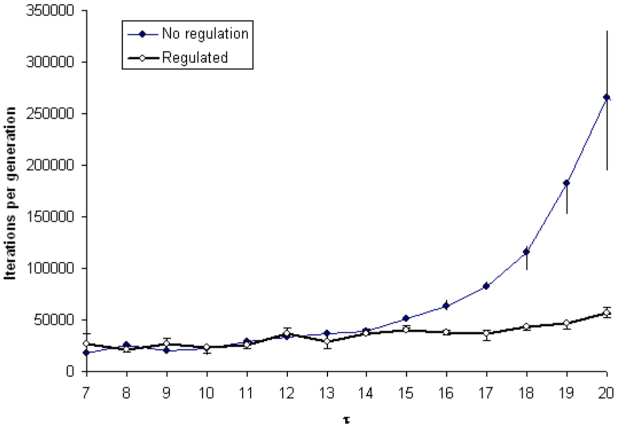

Fig. 6. Effect of division conditions on the advantage of regulation. Data points show median of ten trials, with error bars showing inter-quartile range

the diet switches back from S_2 to S_1. Two further divisions occur towards the end of the second S_1 phase. Stochastic effects can be drastic in these systems. For example, after an accumulation of food due to lack of processing machinery, it is not unusual for a series of divisions over a short time scale to perturb the system so much that the lineage dies out.

B. *Conditions for division*

Figure 6 shows the effect of changing the division count τ on the division rates in regulated an unregulated networks. It can be seen that as τ increases, the difference between division rates of the two networks diverges. Regulated cells divide more quickly when τ is greater than 14. This difference becomes significant when $\tau \geq 16$. This is consistent with the hypothesis that the number of available genes in our network is a limiting factor in growth. There is no analogue of transcription from a single DNA template to multiple mRNA copies of the template in our system. In both substrate conditions, both networks have eight genes available for expression. When $\tau = 16$, new daughter cells have approximately eight B in the metabolome - each gene has an enzyme builder available to carry out expression. At high values of τ, the decay rate of the metabolites outstrips the rate of manufacture. The number of protein builders B rises to τ more rarely because energy is wasted by inappropriate manufacture of E_2.

At low values of τ, the key factor in survival is the complement of metabolites in the daughter cell at division. At these levels of τ it is possible that the entire complement of B can bind to genes that are not critical to survival at the time - enzymes that are needed do not get expressed in sufficient quantities for the cell to survive. Thus the stochastic nature of division drowns out any advantage given by regulation.

C. *Benefits of gene regulation*

Gene regulation should allow cells to grow more quickly, since energy is not wasted in building E_2 when the cell does not need to metabolise S_2. Since our model cells are of fixed radius, we are not in a position to measure growth directly. However, our conditions for death and cell division are indirectly related to the concept of growth in that both are dependant upon the levels of the enzyme builders B. If B has increased to τ, it is likely that levels of the other products of gene expression have also increased. Speed of growth is therefore approximately proportional to the rate of division of cells in the trial.

Diet	Median Division Rate		Probability of a type I error α
	Regulated	Unregulated	
S_1	22.47	4.96	$\alpha < 0.001$
S_2	15.27	4.37	$\alpha < 0.001$
S_1/S_2	18.44	4.19	$\alpha < 0.001$
S_1/starve	14.46	-	–

TABLE III

DIVISION RATES PER MILLION TIME STEPS FOR REGULATED AND UNREGULATED NETWORKS IN THREE DIFFERENT SUBSTRATE ENVIRONMENTS WITH $\tau = 20$ OVER TEN TRIALS. α IS THE PROBABILITY OF ERRONEOUSLY REJECTING THE HYPOTHESIS THAT THE DISTRIBUTIONS OF DIVISION RATES ARE THE SAME FOR REGULATED AND UNREGULATED NETWORKS.

We use the non-parametric Mann-Whitney statistical test for a significant difference to test the distribution of division rates for regulated and unregulated metabolic networks in a range of dietary conditions.

Table III shows the differences in division rates for the ten trials in each environment with and without regulation. In these trials, the division count $\tau = 20$. The significance of the differences between regulated and unregulated networks is also shown in the table. It can be seen that regulation confers higher division rates (and thus competitive advantage) in all environments. Regulation is always beneficial becuase the repressor molecule R is cheaper to manufacture than E_2, with the result that the regulated cells have a better energy budget than unregulated cells. The final row in the table shows division rates where a period of immersion in S_1 is followed by a period of starvation. Here, the unregulated cells failed to divide for seven of the trials, so no meaningful median division rate was calculated.

V. CONCLUSION

By modelling a simplified bacterial metabolome as a set of particles drifting in a fixed, limited space and following a simple set of reaction rules, we have been able to implement a computationally feasible metabolic network whilst preserving sufficient richness to run bio-inspired simulations. We can use this metabolome as a component in a larger experimental framework in which the potential benefits of bacterial genome organisation can be explored in computational and engineering applications of evolutionary algorithms.

It is important to discuss our heuristic selection of the reaction rates for the network. We are aware that we have not fully explored the effect of reaction rates on the efficacy of the network in a rigorous manner. Nevertheless, in the work presented here the regulatory network outperforms the unregulated network with identical reaction rates - and it is likely that it could be made to regulate gene expression even more efficiently should an appropriate (evolutionary) optimisation algorithm be applied to tune the reaction rates.

In our framework, reaction rates are linked directly to the ideas of binding success and binding strength. For an association/dissociation pairing, the binding rate is analogous to the chance of successfully binding, and the binding strengh is analogous to the chance of dissociating. If binding is considered as some (potentially multidimensional) inexact string matching process then it can be seen that binding rates and strengths can be evolved appropriately. A key discovery in this paper is a feel for the range of binding values evolutionary systems should deliver. Association and dissociation rates in the network presented here span three orders of magnitude, and this range is critical for appropriate regulation. For evolutionary exploration of the network dynamics, an appropriate representation of binding is therefore needed - one in which we will get the distribution of binding rates and strengths that we have found to be necessary. If binding is allowed to vary such that very low bind strengths emerge, the potential to build new networks between metabolites becomes apparent.

We also have a better feel for the numbers of agents required to run a metabolic model that can be subjected to evolutionary pressure. This is controlled by our division count τ. We have found that where τ is very low, the stochasticity of the cell environment drowns out any advantage that gene regulation confers on the system. However, we have found a range of computationally tractable levels of τ where gene regulation does offer a competitive advantage.

Our network is regulated via a highly simplified model of gene expression that is more similar to regulation at translation than regulation at transcription. Regulation at transcription is more effective - a single repressor molecule blocks the DNA and prevents transcription to mRNA which in turn would be used to build thousands of enzymes. Thus transcriptional repressors are cheap and very effective. In contrast, a single translational repressor prevents the creation of far fewer enzymes and is less efficient. Our implementation is more akin to an RNA-world model, in which there is no transcription and regulation can only occur at the translation stage. This is not the only similarity between our model and models of life forms on the early Earth [14], in that the individual is little more than a “bag” of semi-autonomous genetic elements, genes are small and freely available when unregulated, genetic multiplicity is preserved, and cell constituents are distributed randomly at cell division. This scenario offers many avenues for research. We are particularly attracted to the idea that lateral gene transfer can be used to spread successful responses from an individual to the local population.

The two dimensional organisation of our model has allowed us to simulate the random mixing of the components of the metabolome via a simple motion model. This has been computationally expensive, particularly when calculating binding reactions, where two components need to be sufficiently close - an n^2 set of calculations is needed to find which agent is closest. Although there are possibilities for improving the spatial model, we intend to move to an aspatial model of the metabolome and model spatial mixing stochastically to increase computational efficiency. This would be compatible with an object oriented model of the bacterial metabolome [15], one that can form part of a broader model of bacterial evolutionary mechanisms.

References

[1] T. Toffoli, “What you always wanted to know about genetic algorithms but were afraid to hear,” in *Perspectives on Adaptation in Natural and Artificial Systems—Essays in honor of John Holland*, L. Booker, Ed. Oxford University Press, 2004.

[2] K. Deep and M. Thakur, “A new crossover operator for real coded genetic algorithms,” *Applied Mathematics and Computation*, vol. 188, no. 1, pp. 895–911, 2007.

[3] M. G. Pires, I. N. da Silva, and F. C. Bertoni, “Solving shortest path problem using Hopfield networks and genetic algorithms,” in *Hybrid Intelligent Systems*. Los Alamitos, CA, USA: IEEE Computer Society, 2008, pp. 643–648.

[4] D. S. Burke, Kenneth, J. J. Grefenstette, C. L. Ramsey, and A. S. Wu, “Putting more genetics into genetic algorithms,” *Evolutionary Computation*, vol. 6, no. 4, pp. 387–410, Winter 1998. [Online]. Available: citeseer.nj.nec.com/burke98putting.html

[5] H. Kitano, “Systems biology: a brief overview.” *Science*, vol. 295, no. 5560, pp. 1662–1664, March 2002.

[6] K. N. Houk and P. H. Cheong, “Computational prediction of small-molecule catalysts,” *Nature*, vol. 455, pp. 309–313, 2008.

[7] D. T. Gillespie, “Exact stochastic simulation of coupled chemical reactions,” *The Journal of Physical Chemistry*, vol. 81, no. 25, pp. 2340–2361, 1977.

[8] M. Gibson and J. Bruck, “An efficient algorithm for generating trajectories of stochastic gene regulation reactions,” Caltech Parallel and Distributed Systems Group [http://caltechparadise.library.caltech.edu/perl/oai2] (United States), Tech. Rep., 1998.

[9] J. S. van Zon and P. R. ten Wolde, “Green’s function reaction dynamics: a new approach to simulate biochemical networks at the particle level and in time and space,” 2004. [Online]. Available: http://www.citebase.org/abstract?id=oai:arXiv.org:q-bio/0404002

[10] D. Bray, M. D. Levin, and K. Lipkow, “The chemotactic behavior of computer-based surrogate bacteria,” *Current Biology*, vol. 17, pp. 12–19, 2007.

[11] I. D. Couzin, J. Krause, R. James, G. D. Ruxton, and N. R. Franks, “Collective memory and spatial sorting in animal groups.” *J Theor Biol*, vol. 218, no. 1, pp. 1–11, 2002 Sep 7.

[12] L. You, “Toward computational systems biology.” *Cell Biochem Biophys*, vol. 40, no. 2, pp. 167–184, 2004.

[13] A. J. F. Griffiths, S. R. Wessler, R. C. Lewontin, and S. B. Carroll, *Introduction to Genetic Analysis*, 9th ed. W. H. Freeman, 2007.

[14] C. Woese, “The universal ancestor,” *Proceedings of the National Academy of Sciences of the United States of America*, vol. 95, no. 12, pp. 6854–6859, 1998.

[15] K. Webb and T. White, “UML as a cell and biochemistry modeling language.” *Biosystems*, vol. 80, no. 3, pp. 283–302, June 2005.