Achieving a Simple Development Model for 3D Shapes: Are Chemicals Necessary?

Pauline C Haddow
CRAB lab
Department of Computer and Information Science
NTNU, Trondheim, Norway
pauline@idi.ntnu.no

Johan Hoye
CRAB lab
Department of Computer and Information Science
NTNU, Trondheim, Norway

ABSTRACT

Artificial Development Systems have been introduced as a technique aimed at increasing the scalability of evolutionary algorithms. Most commonly the development model is part of the evolutionary process, each individual developed during fitness evaluation. To achieve scalability it may be argued that the implicit requirements of evolvability and effectivity (in terms of its resource requirements) are thus placed on the development model. To achieve an effective development model, one of the challenges is to find appropriate mechanisms from developmental biology and ways to implement them for the application in hand. This work presents a development model for the evolution and development of 3D shapes. The goal being to create a simple development model for any 3D shape. Further, this work provides a preliminary investigation into the usefulness of one of the mechanisms implemented in this model, that of chemicals.

Categories and Subject Descriptors

I.2.M [**Artificial Intelligence**]: [Miscellaneous]

General Terms

Experimentation

Keywords

Artificial Development, 3D Shapes, Biological Mechanisms, Chemicals

1. INTRODUCTION

One recognised limitation of Evolutionary Algorithms (EAs) is known as the scalability problem. For many problems, as the problem scales up, EAs become too computationally expensive to be useful. A number of approaches aimed at reducing the scalability problem have been proposed over the years. Refinements to the evolutionary algorithms themselves include, but are not limited to: Messy Genetic Algorithm [8]; fast Messy Genetic Algorithm [9]; Linkage Learning Genetic Algorithm [10] and the Breeder Genetic Algorithm [17].

Other approaches attempt to simplify the problem itself before the application of evolution. These approaches apply a refinement of the well known design technique : divide and conquer such as in [13], [19] and [20]. The results in this area are promising, but the automatic identification of independent sub-problems is tricky and the methodology infers substantial overhead.

A further approach is to simplify the description of the problem when presented to evolution. Biological development is nature's way of coping with scaling. If one considers a mature human against a mature mouse, the difference in size and complexity of functionality are tremendous. However, the difference in size of their respective genome is relatively small due to the fact that the genome does not contain explicit information about every detail of the body. Instead it serves as a building plan, describing how the organism is to be built. As the vice president for medical research at Howard Hughes Medical Institute puts it, "complexity does not come from the number of genes but from the way in which they are used" [7].

In an artificial development approach, the problem is represented as DNA that may be decompressed (developed) into a potential solution. Evolutionary algorithms may be applied to the DNA style representation to search for a suitable DNA for the problem in hand. Each selected individual of the evolutionary process is developed into its respective solution (organism) so as to evaluate the fitness of the individual.

Simplifying the problem in the form of DNA does not in itself solve the scalability problem. The DNA may be a simpler description than that of a corresponding direct representation but the regulation of the genes in the DNA ,resulting in the development of the organism, is a complicated process. As such, one might say that a simpler genotype is being evolved but at the cost of a more complicated evaluation phase.

There have emerged two clear views as to how to approach the design of artificial development systems: biologically plausible and biologically inspired. Biologically plausible models seek to model biological development in its entirety, examples of which may be seen in [1], [4] and [14]. Expansion of these models and other similar models will lead

GECCO'07, July 7–11, 2007, London, England, United Kingdom.

to a greater understanding of the workings of developmental systems, an area of biology where there is much to be learned.

This work focuses on bio-inspired development, drawing inspiration from biological development but not attempting to model/copy the process in its entirety. The main goal being to design an effective developmental model, where the DNA may be tuned to achieve scalable solutions. Examples of bio-inspired models may be found in [3], [6], [16] and [18].

A number of attempts to achieve 2D structures through the application of development may be seen in the literature, the most frequently seen structure being that of the French Flag e.g. [16, 21]. Two examples of excellent work resulting in modelling of 3D shapes or organisms are the work of Eggenberg [5, 12] and Hogeweg [11]. Both these developmental models are relatively complex models that may be classed under biologically plausible models.

The work of Bongard and Pfeifer [2, 3] takes a simpler, bio-inspired approach to 3D organisms. In their model, morphogenesis is present in the form of changes in cell size. As a cell (unit) grows in size, a divide threshold limits further growth within the cell forcing the cell to divide into two new cells. Each cell has diffusion sites on each connection to its 6 neighbours, enabling diffusion of chemicals both between the connections of a given cell and to neighbouring cells. The phenotype sought is the morphology and neural control of simulated agents.

In this work, a bio-inspired model for 3D shapes is presented. Morphogenesis within a single cell and chemical diffusion are not included in the model. However, proteins are included, in addition to chemicals. The model is also aimed at achieving flexibility with respect to the 3D shapes that may be developed. As in Bongard, a variable length genetic algorithm is the evolutionary algorithm applied.

To understand whether developmental models can in fact scale, it is important to understand how to design them. There is little in the literature today to really help one design an artificial developmental process apart from the fact that perhaps growth and specialisation are features present in most models. In general, appropriate mechanisms from biological development need to be identified together with efficient ways in which these may be represented so as to create both evolvable DNA representations and scalable solutions.

One mechanism of biological development that may be seen in many developmental models, including the one presented herein, is the use of chemicals. The work of Miller [15] showed that for the French flag problem and the model of development presented therein — which includes chemical diffusion; an increase in the number of chemicals has a direct positive effect on fitness. However, is such a mechanism always necessary?

This paper provides further investigation into the use of chemicals in developmental models by conducting a preliminary investigation into the usefulness of chemicals with respect to 3D shapes, in the context of the development model presented herein.

Section 2 provides a description of the development model created in this work. In section 3, the experiments investigating the development of 3D shapes and the usefulness of chemicals in the model are presented. Finally, in section 4 the conclusions of this work are presented.

2. THE MODEL

An organism is a three dimensional grid with a fixed size. A location inside the organism may be either empty or filled by a cell and a given cell has a state. Each cell has a fixed size and may occupy exactly one location within the grid and only one cell is allowed at each location. A maximum number of cells are allocated to the organism.

Inside each cell is a DNA (string of genes); a number of proteins and a number of chemicals. The number of chemicals is constant for all cells and chosen for a given experiment but the level of each chemical is allowed to vary. Each cell is of a specific type which is one of a set of explicitly defined cell types. In the experiments herein a type refers to a colour.

The genotype form of an individual is converted to a DNA for the development process. The number of development steps allowed and the bounds (3D grid) of the organism sought are given to the development process as well as the DNA of the individual being developed. Evolution is applied to tune the individuals, resulting in a fit DNA which in turn results in the developed solution.

2.1 Variable Length Genetic Algorithm

A variable length Genetic Algorithm using tournament selection; two-point crossover; crossover rate of 0.9 and mutation rate of 0.1 (divided into 2 levels of mutation) is applied. At the genotype level, mutation is applied through gene duplication or gene removal. The probability of one of these events is 0.025. The new gene created under gene duplication is added at the end of the genotype. When a gene is removed, the remaining genes are spliced together at the point of removal. The second level of mutation is at the gene level and varies depending on the type of terminal value. For terminals represented as strings — a bit is flipped, and for terminals represented as counters (integer) — either the value is increased or decreased by a single step or the value is swapped from the set of possible values.

The genotype consists of a number of genes. Each gene has a promoter and coding region. The promoter contains the promoter ID, which is used when deciding which genes are to be transcribed. The coding region contains all information necessary to construct the protein the gene codes for: a precondition, its function and time to live.

The precondition of the protein consists of two parts. The first part checks the cells neighbourhood. A neighbourhood pattern consists of 6 values where each neighbour is either represented by cell type, no cell or don't care. It should be noted that the cell itself is not included in this calculation. The second part, the chemical part, specifies the concentration threshold for each chemical in the cell. As stated, the number of chemicals in the cell is constant for a given simulation. As such, the precondition is fulfilled if the state of the neighbours matches the neighbourhood pattern and if all the cell's chemical levels exceed the specified thresholds.

A protein can have one of the following functions: divide cell, change chemical concentration (produce or consume chemicals), change cell type and transcribe genes. However, a protein is not able to directly perform any actions, it can only request the cell it resides in to perform the action for it. If and how the action is actually performed, is decided by the cell. This way of handling the protein actions eliminates the problem of deciding in which order the proteins should be allowed to perform their actions. The time to live

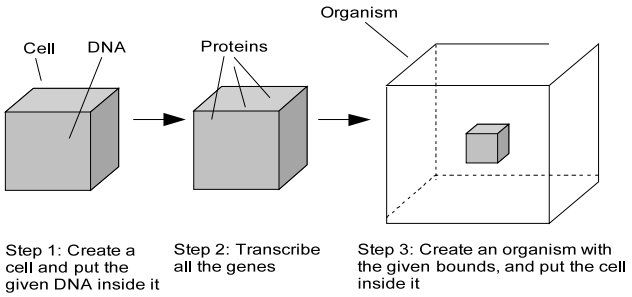

Figure 1: Initialisation of the Development Process

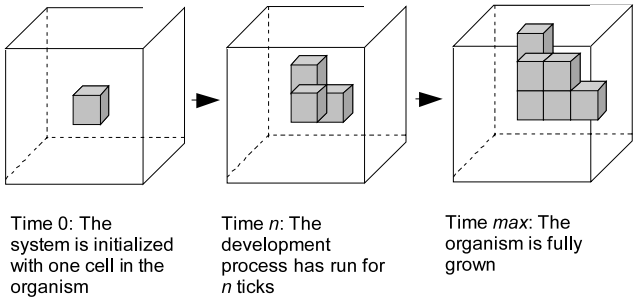

Figure 2: Development of the Organism (3D shape)

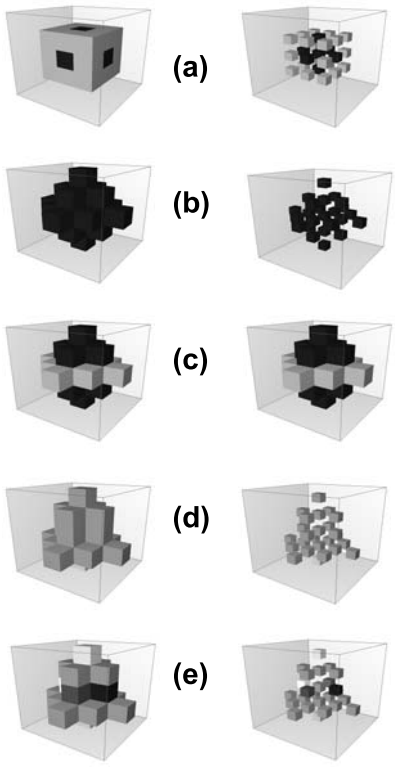

Figure 3: Targets: (a) cube with a cross, (b) tree, (c) xmas tree, (d) sphere and (e) divided sphere. Normal(left) and Exploded View(right)

specifies how many development steps the protein is active before it dies and is removed from the cell.

Fitness is normalised based on a maximum fitness of 2N for the N blocks allocated to the organism. If a block is correct then fitness is increased by 2; if it is of the incorrect type then fitness is increased by 1 or if it is in the incorrect place then no fitness credit is given.

2.2 Overview of the Development process

The development process is accomplished in two steps: initialisation and development. First, an empty cell is created. The DNA, given as input to the development process, is placed inside this cell and all the genes in the DNA are transcribed resulting in a number of proteins in the cell, see step 2 in figure 1. Then, an organism is created using the given bounds and the cell is placed in the middle of this organism, as illustrated. The development process is now initialised.

The process of development is divided into discrete time steps. At a given step, the same series of discrete events occur: the chemical concentrations are adjusted; the proteins are notified that a step has occurred and the cell performs the actions requested by the proteins. As such, each step is a series of 3 events for a single cell in the organism.

Event 1: Adjusting the chemical concentrations involves reducing or increasing illegal values to the nearest legal bound. As such, this is a practical, non biological event.

Event 2: The protein updates involve requesting the actions described by the functionality of those genes whose protein preconditions are met i.e. the proteins that are transcribed.

Finally in event 3: The actions requested for the given cell, are performed in a prioritised order: transcribe genes; change chemical concentrations; divide cell and then change the cells type.

This description is for the process in a single cell. In nature, the cell update across the organism is a highly parallel process. However, in this model, cell update is conducted in a sequential manner following the order in which cells were created, starting with the zygote. Further, the mechanism of contact inhibition is implemented in the model. If a cell grows into a neighbouring cell, this cell is then active and no other cell can grow into this cell.

Protein updates may be said to be highly parallel as at the end of this event the protein actions are still only requests and have no effect on each other's request. As such, the order that they have been processed has no effect on the resulting requests. Separating the protein updates from the cell updates in this way, enabled a certain amount of parallelism whilst retaining a simpler model.

Further the cell actions themselves are performed in a priorised manner: transcribe genes, change chemical concentrations, divide cell and then change cell's type. In the transcription of the genes, all requests (from the proteins) for a gene transcription action are conducted. Since several genes may request a chemical concentration change — increase or decrease in concentration, a given chemical may acquire a chemical value outwith its bounds thus requiring the illegal value adjustment described.

When a protein requests a division it provides a positive, negative or neutral stimulus for growth in each of the 6 possible directions. When all protein divide requests for the cell are processed, the total stimulus for each direction is pro-

cessed. The cell may divide in up to 6 directions and each division is dependant on whether that particular direction's stimulus is over a given threshold.

Changing the cell type was given a similar stimulus procedure. However, here a winner is chosen being that with the greatest stimulus level. If this level is over the threshold then the cell type is changed to the given type.

The development process continues until the organism has developed for the given number of steps. The fully grown organism is then generated as output of the system. An example of a developing organism is given in figure 2. It should be noted that in the development system development doesn't stop when the sought after shape is achieved. The development process is an ongoing process and the goal is to achieve a stabile phenotype despite the fact that the process is ongoing.

3. EXPERIMENTS

The experimental setup for evolution was as follows: population size 1000, crossover 0.9, mutation rate 0.1, maximum generations 500 and tournament selection with a group size of 4. Fitness was measured on a cell by cell comparison and normalised based on a maximum value of 2N for N cells.

The development setup involved a 6 bit promoter, 5 initial genes in the DNA, 3 proteins types (divide cell, transcribe gene and change type), a 5 step time to live and a selected number of chemical types from 0 to 10. Development time was 12 steps and a 5x5x5 grid was allocated to the organism. The experiments were run on a Beowolf cluster.

3.1 Achieving Growth, Differentiation and Morphogenesis

In this set of experiments, each experiment was run 20 times and the fitness, the DNA and the resulting phenotype were stored for the best individual and the standard deviation on the resulting fitness calculated. This set of experiments aimed to develop simple 3D shapes requiring growth, differentiation and morphogenesis. Differentiation is highlighted by the different cell types (cell colours) required and morphogenesis is limited to growth restriction. Cell movement, cell shaping and cell death are not implemented in the model.

Five phenotypes were designed to be used as targets: a cube with a 3D cross inside; a tree; a xmas tree; a sphere and a divided sphere. For ease of viewing of the inside cells, an exploded view of the shapes is used to display the results. Figure 3 illustrates the shapes sought and their respective exploded view.

Figure 4 illustrates the development steps needed to create a simple sphere (the target shown), starting with a single cell. At time 2 i.e. after 2 development steps, the sphere is already fully developed. Further, although the development process continued for another 10 steps, no change may be seen in the organism.

Figure 5 presents a similar sphere but this time involving 2 different cell types i.e. 2 different colours. Perhaps not surprisingly, the introduction of specialisation (change of type) slows down development but with such a simple shape and only two cell types the organism may still be developed in few steps i.e. 5 in this case. The developed shape, again, remained steady a number of steps. The cube has also 2 cell types and, similarly, is developed in 5 steps and remained steady a number of steps (see figure 6).

As illustrated in figures 7 and 8, the development model was more challenged by the tree and christmas tree shapes. In neither case did the best individual achieve 100% fitness. As shown, development achieves the structure of the organism but not the correct specialisation. What is interesting is that the decorations on the xmas tree are in the correct place i.e. the dark cells, but they are all dark instead of the sought colour. But why dark the colour of the zygote? Also, it is noted that a main colour(type) is introduced early in the development process and from this point no new types are introduced. However, the simpler tree shape only requires two types but introduces a third type (light) during development which then reverts to the correct colour. This additional type may be introduced as a step towards achieving the final type required. Further, although hard to see, the base of the tree requires a fourth type. This type is not achieved. The reasoning for this differentiation is unclear at this point. However, further investigation is needed to ascertain whether the developmental model would achieve such specialisation with sufficient development steps.

3.2 Effect of Chemicals on the Developed Organisms

It was interesting to consider whether increasing the number of chemicals might support or hinder development of these shapes. Figure 9 illustrates the effect of increasing the number of chemicals available to development for the simple case of a sphere to a divided sphere and finally to the christmas tree. For the case of the sphere, increasing the chemicals from 0 to 5 has little effect. However, some deviation in the results may be seen. However, suddenly at 10 chemicals fitness dropped quite drastically. Moving to the divided sphere, increasing the number of chemicals seems to decrease fitness. However, it should be noted that the standard deviation is quite large. Moving on to the christmas tree, one can say that this is a more challenging shape and requires a more specialisation. The trend of reducing fitness with increasing chemicals is clear but again with a significant standard deviation in the results. This trend was also present in the other shapes.

These results would indicate that there seems to be some sort of connection between decreasing fitness with increasing chemicals and that this effect is more pronounced as the difficulty of the shape and specialisation increase. With the increasing standard deviation seen as the number of chemicals increases this might indicate that although a high fitness may occur with higher number of chemicals, this is much less likely than with lower levels of chemicals.

3.3 Effect of chemicals on 3D shapes in general

To study the effect of chemicals on 3D shapes, in general, it was not realistic to study all possible shapes. Instead, it was chosen to concentrate on two shape parameters: that of symmetry and that of colour. The use of symmetry may be said to be a way of classifying the type of shapes that may be developed. It is also a parameter that highlights how good the system is at achieving growth and morphogenesis. The second parameter, colour represented as cell type, provides a challenge for the system with respect to increasing specialisation.

The number of chemicals investigated was 0,1,2,4 and 8. For each chemical number, four shapes were investigated:

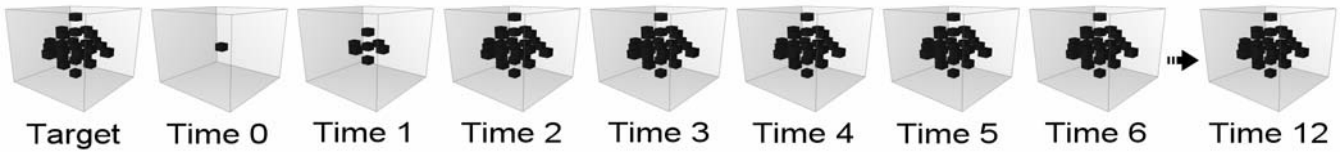

Figure 4: Development of a Sphere without Specialisation

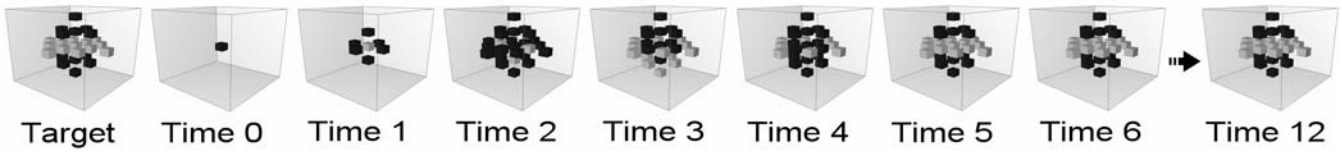

Figure 5: Development of a Sphere with Specialisation

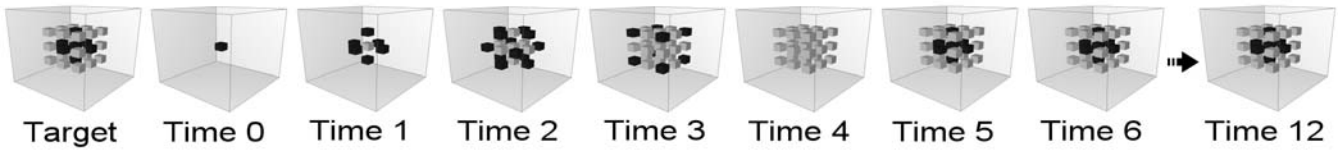

Figure 6: Development of the Cube

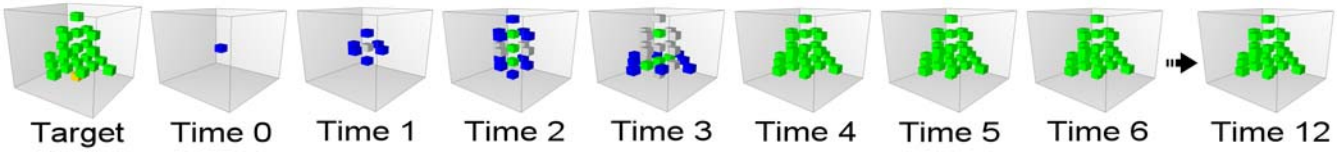

Figure 7: Development of the Tree

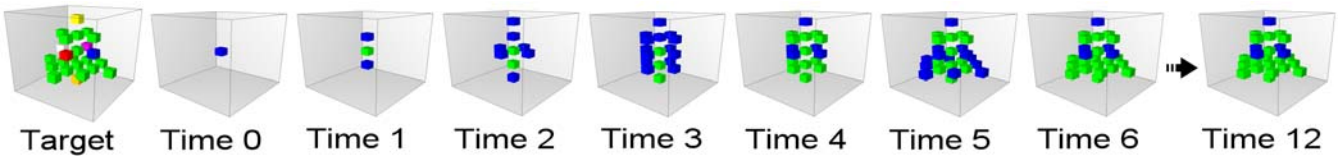

Figure 8: Development of the Christmas Tree

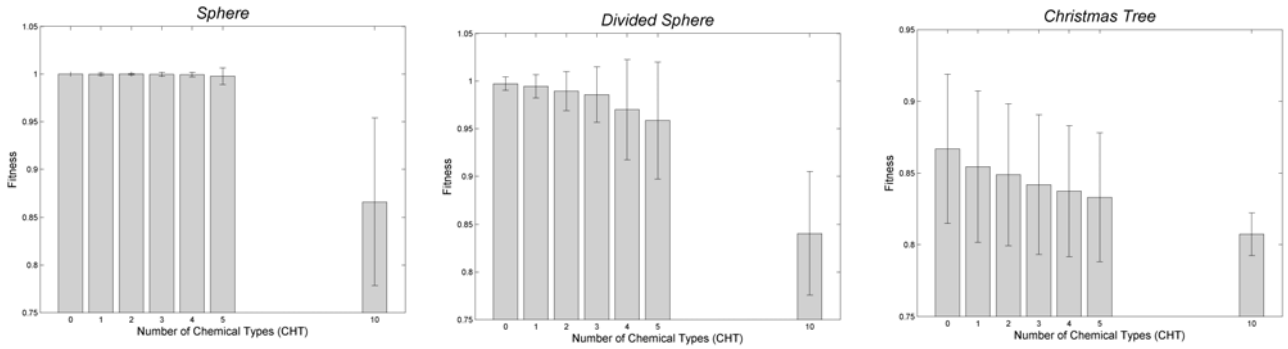

Figure 9: Increasing the Number of Chemicals: Sphere, Divided Sphere and Christmas Tree

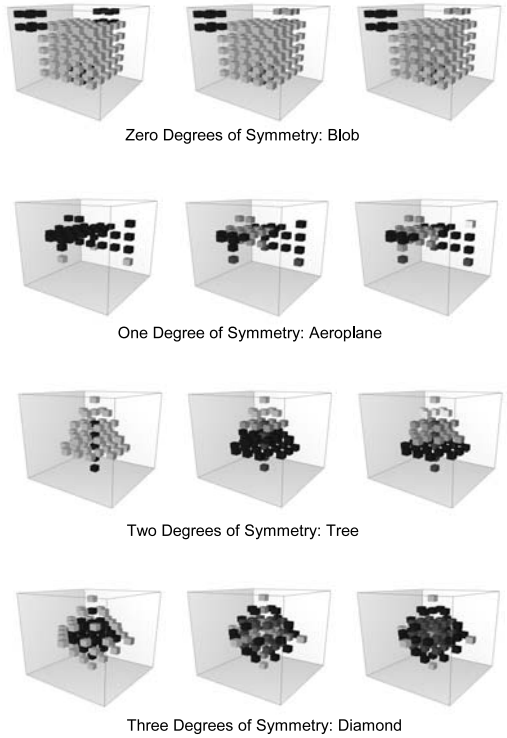

Figure 10: Degrees of Symmetry: with 2 (left), 4 (Centre) and 6 (right) Cell Types

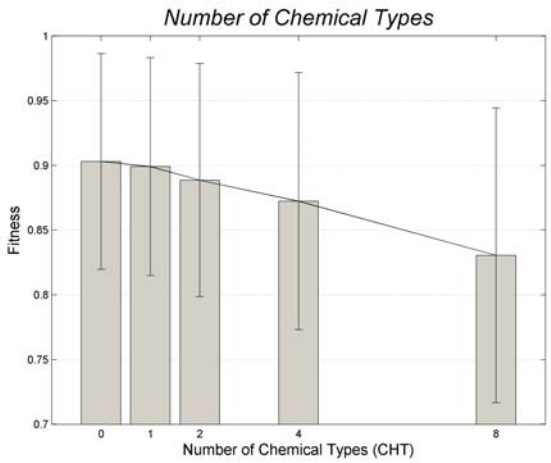

Figure 11: The Effect on Fitness of Varying Number of Chemicals for the 12 cases

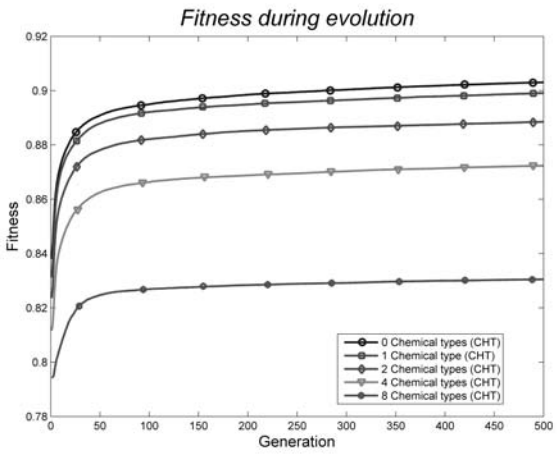

Figure 12: Fitness Variation during Evolution when Varying the Number of Chemicals for the 12 cases

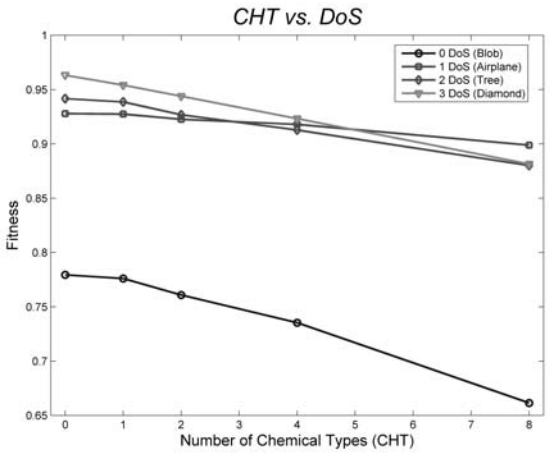

Figure 13: The Effect on Fitness for each of the 4 Key Shapes whilst Varying the Number of Chemicals

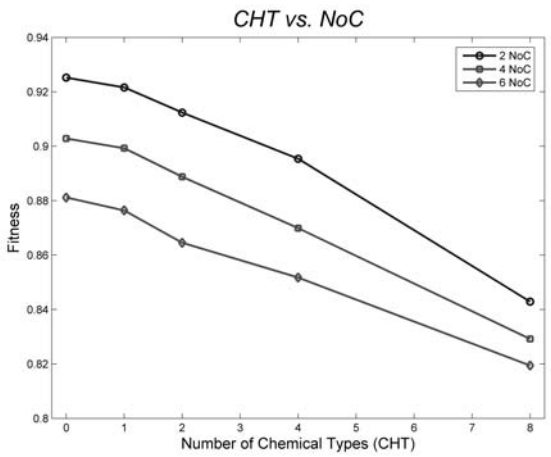

Figure 14: The Effect on Fitness for each of the Number of Colours whilst Varying the Number of Chemicals

a blob (0 degrees of symmetry (DoS)); an airoplane (1 DoS); a tree (2 DoS) and a diamond (3 DoS) as illustrated in figure 10. Each shape was further investigated for 2,4 and 6 colours. As such, 12 experiments were conducted for each chemical number and all experiments were repeated 20 times.

It was hypothesised that increasing the symmetry of the shape reduces the difficult of developing the shape. This may not be so unexpected as the developmental system was designed to grow from the centre out, thus a more symmetric shape would be easier to achieve. Increasing the number of colours increases the choice of cell types for any cell to specialise into, thus increasing the search space of possible solutions. Results, however, were not so conclusive. It would seem that the asymmetric nature of the blob provided the system with quite a challenge. However, for the symmetric cases, there was only a slight improvement in the average fitness for increasing symmetry.

Figure 11 illustrates the average fitness achieved for all shapes and each number of colours per number of chemicals. As shown, there is a clear trend for decreasing fitness for increasing number of chemicals over the 12 cases. In fact, there is an almost linear decrease in fitness with increasing number of chemicals i.e. a negative correlation between fitness and the number of chemicals. To test this impression for statistical significance a one-way ANOVA test was employed to compare group means. The results obtained from testing the data at a 95% confidence interval strengthens the assumption that there is in fact a negative correlation between fitness and the number of chemicals. It showed that there are indeed statistically significant differences amongst the group means.

It was interesting to look at fitness throughout evolution for each chemical experiment to see if the negative correlation was consistent throughout evolution. As shown in figure 12, 0 chemicals is in fact the most effective solution throughout evolution and the negative correlation appears through evolution. This is, perhaps not so surprising as increasing the number of chemicals also increases the search space for the solution. The results also indicate a clear trend that, for each of the results in the figure, fitness increases quite sharply at the start and slowly increases for the remainder of the 500 generations.

A further question that may be raised from figure 12 is why fitness is so low on average. Figure 13 illustrates that it is the blob, the zero degree of symmetry shape that pulls down the average fitness significantly. However, if one considers that this shape is asymmetric and all the others are symmetric then this result is not so surprising. However, what is perhaps more surprising is that the individual shapes paint a different picture of the effect of chemicals than the average fitness case. Changing the number of chemicals for the 1 DoS shape has little effect on the average fitness. However, the 1 DoS shape has the lowest fitness for few chemicals and the highest fitness for the maximum number of chemicals. It may be the case that the blob provides a very difficult problem for evolution and thus evolutionary search becomes more random. For the symmetric shapes there does not seem to be an advantage with increasing the number of chemicals but the disadvantage seems to be less for lower degrees of symmetry.

Finally, the effect that the number of colours has on this negative correlation was investigated. As shown in figure 14, the three lines representing the 3 colour experiment sets are monotonically decreasing, almost running perfectly parallel.

4. CONCLUSION

The developmental model presented herein has been shown to be flexible, with respect to the type of 3D shapes that may be produced although some shapes have been shown to be harder than others. Also from the average fitness results it is obvious, within the constraints given to the developmental process and the evolutionary process, that any of the 3D shapes are guaranteed to be developed. However, the work has highlighted that these shapes may be achieved without chemical dispersion and a relatively stable organism is achieved. However, whether dispersion can speed up the results sought or whether dispersion might lead to fault tolerance to external influences has not been investigated.

Further the effect of increasing the number of chemicals has been investigated with respect to 3D shapes of zero to three degrees of symmetry and with 2, 4 and 6 types. The results indicate that there is a negative correlation between the fitness achievable and the number of chemicals applied. In the light of earlier work by Miller [15] which presented a positive correlation between fitness and number of chemicals, this result is surprising. However, this work does not in any way invalidate the work of Miller. There are distinct differences between the developmental models and the applications sought. These results should be seen in the light of the model itself and the application goal. There may be factors in the model that affect these results other than those currently investigated. Further, investigation should involve comparison between different mechanisms and properties of the developmental system and between different applications.

5. REFERENCES

[1] P. Bentley and S. Kumar. Three ways to grow designs: A comparison of embryogenies for an evolutionary design problem. In *Proceedings of the Genetic and Evolutionary Computation Conference*, volume 1,

pages 35–43, Orlando, Florida, USA, 13-17July 1999. Morgan Kaufmann.

[2] J. Bongard and R. Pfeifer. Repeated structure and dissociation of genotypic and phenotypic complexity in artificial ontogeny. In *Proceedings of the Genetic and Evolutionary Computation Conference*, pages 820–836. Morgan-Kaufmann, 2001.

[3] J. Bongard and R. Pfeifer. Evolving complete agents using artificial ontogeny. In F.Hara and R. Pfeifer, editors, *Morpho-functional Machines: The New Species (Designing Embodied Intelligence)*, pages 237–258. Springer-Verlag, 2003.

[4] F. Dellaert. Toward A biologically defensible model of development. Master's thesis, Nov. 1995.

[5] P. Eggenberger. Evolving morphologies of simulated 3d organisms based on differential expression, 1997.

[6] D. Federici. Evolving complete agents using artificial ontogeny. In *Proceedings of SAB, Simulation of Adaptive Behavior*, pages 373–384, 2004.

[7] e. a. G. M. Rubin. Comparative genomics of the eukaryotes. In *Science*, pages 2204–2215. 2000.

[8] D. Goldberg, K. Deb, and B. Korb. Messy genetic algorithms: Motivation, analysis, and first results. *Complex Systems*, 3:493–530, 1989.

[9] D. E. Goldberg, K. Deb, H. Kargupta, and G. Harik. Rapid accurate optimization of difficult problems using fast messy genetic algorithms. In S. Forrest, editor, *Proc. of the Fifth Int. Conf. on Genetic Algorithms*, pages 56–64, San Mateo, CA, 1993. Morgan Kaufmann.

[10] G. R. Harik and D. E. Goldberg. Learning linkage. In R. K. Belew and M. D. Vose, editors, *Foundations of Genetic Algorithms 4*, pages 247–262. Morgan Kaufmann, San Francisco, CA, 1997.

[11] P. Hogeweg. Shapes in the shadow: Evolutionary dynamics of morphogenesis. In *Artificial Life*, volume 6, pages 85–101, 2000.

[12] P. E. Hotz. Comparing direct and developmental encoding schemes in artificial evolution: A case study in evolving lens shapes. In *Proc. of the 2004 IEEE Congress on Evolutionary Computation*, pages 752–757, Portland, Oregon, 20-23 June 2004.

[13] T. Kalganova. Bidirectional incremental evolution in extrinsic evolvable hardware. In J. Lohn, A. Stoica, and D. Keymeulen, editors, *Proceedings of the Second NASA/DoD Workshop on Evolvable Hardware (EH2000)*, pages 65–74, Palo Alto, California, USA, 13–15 2000. IEEE.

[14] S. Kumar and P. J. Bentley. Implicit evolvability: An investigation into the evolvability of an embryogeny. In *Late Breaking Papers in The Second Genetic and Evolutionary Computation Conference (GECCO 2000)*, 2000.

[15] J. F. Miller. Evolving a self-repairing, self-regulating, french flag organism. In *Proceedings of the Genetic and Evolutionary Computation Conference*, pages 129–139, 2004.

[16] J. F. Miller and W. Banzhaf. Evolving the program for a cell: from french flags to boolean circuits. In S. Kumar and P. J. Bentley, editors, *On Growth, Form and Computers*. Academic Press, Oct. 2003.

[17] H. Muhlenbein and D. Schlierkamp-Voosen. Predictive models for the breeder genetic algorithm: I. continuous parameter optimization. *Evolutionary Computation*, 1(1):25–49, 1993.

[18] G. T. P.C. Haddow and P. van Remortel. Shrinking the genotype: L-systems for ehw? In *Proceedings of International Conference on evolvable systems: from Biology to Hardware*, pages 128–139, 2001.

[19] R. C. Purshouse and P. J. Fleming. An Adaptive Divide-and-Conquer Methodology for Evolutionary, Multi-criterion Optimisation. In C. M. Fonseca, P. J. Fleming, E. Zitzler, K. Deb, and L. Thiele, editors, *Evolutionary Multi-Criterion Optimization. Second International Conference, EMO 2003*, pages 133–147, Faro, Portugal, April 2003. Springer. Lecture Notes in Computer Science. Volume 2632.

[20] J. Torresen. A divide and conquer approach to evolvable hardware. In *Proc. of the Second Int. Conf. on Evolvable Systems: from Biology to Hardware (ICES98)*, 1998.

[21] L. Wolpert. In *Principles of Development*. Oxford University Press, 1998.