Investigating the effect of Regulatory Decisions in a Development Model

Pauline C Haddow
and Johan Hoye
CRAB lab, Department of Computer and Information Science Email: pauline@idi.ntnu.no, johan@idi.ntnu.no

***Abstract*—Artificial development has been introduced by many as a means to simplify the genome of the evolutionary process and thus aid scalability of evolutionary techniques. However, this simplicity in the genome comes at the cost of complexity in the mapping. This is perhaps not so surprising when we look to biology and the complicated process of gene regulation. However, creating an artificial representation of this complicated process is far from straight forward. To simplify such a process, we need to acquire knowledge and define some form of rules to guide the creation of development models. The work presented herein investigates an existing development model, identifying which factors in the model are part of the regulatory decisions. Further, experimental work looks more closely at protein pre-conditions within the model. The results form the basis for more generalised preliminary rules for protein pre-conditions creation.**

I. Introduction and Motivation

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" [1]. So how are genes being used in artificial development?

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. 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 development of the organism, is a complicated process.

Many attempts have been made to design development algorithms (development models) to meet the needs of different applications i.e. agent systems [2], [3]; electronic circuits [4], [5], [6], [7], [8], [9]; neural networks [10], [3]; and 2D and 3D shapes [5], [11], [12], [13], [14]. Others aim to replicate the biological process of development so as to gain a better understanding of such a process [15], [16], [17]. Further, some work has begun to investigate mechanisms within development models [18], [19], [20] and characteristics of such models e.g. robustness, scalability [21], [22], [18], [23], [24] so as to learn more about how to apply development. However, *we are still far from understanding what a development model is, how it should be applied and what such a model might achieve*.

There is much interesting ongoing work within development. However, a challenge is to know where to begin and what to do next. It would, perhaps seem easiest to start with an existing developmental model, applied by a known author in the field, and try to refine the model to the needs of your own application. However, when these models are not fully understood, it may be the wrong starting point and thus success is unlikely. On the other hand, you might wish to create your own model from scratch. In the early stages of a field, the latter may be said to be a very viable approach. However, we have no rules describing how to approach the creation of a development model, apart from the fact that many development models have some form of growth and specialisation, but not all. To find the rules, we need to understand the effect of decisions made by authors or decisions that you yourself might want to make.

In earlier work [19], the development model applied herein was presented and a number of shapes were developed, illustrating the flexibility of the model with respect to different shapes and different colours (cell types). Such results were shown to be achieved without chemical dispersion and were shown to result in stable organisms —- the development process continued with no effect on fit individuals. The work also drew attention to the effect of chemicals in the model and particularly the negative correlation between increasing chemicals and fitness. Although this result was in direct contradiction to the work of Miller [18] it was also acknowledged that such results are likely to be due to the differences between the development models themselves and the applications sought.

The work presented herein takes a closer look at the model in the light of the regulatory decisions made. Experimental work is focused on investigating the relationship between changes applied to protein preconditions in the model and the phenotypes achieved. This work forms the basis for more generalised preliminary rules for the creation of such protein preconditions in a model.

Section II provides an overview of the elements in today's development models which may be said to play either a direct or indirect role in gene regulation. The development model under investigation is presented in section III, including a detailed description of the gene regulation built into the model. In section IV, a more detailed description of the protein pre-conditions in the model is provided and section V presents the experimental work investigating such pre-conditions. Finally section VI, concludes the experimental work and provides a more general conclusion with respect to rules for pre-conditions in such models.

II. Gene Regulation in a Development Model

Gene regulation may be achieved at different levels and in different ways. It is, therefore, important to identify what factors in a particular development model are infact a part of the gene regulatory decisions.

Gene regulation may involve explicit and implicit features controlling the regulation of the genes. Genes may regulate other genes through their protein products. Such regulation may, unlike in biology, allow one protein to regulate more than one gene. Such a decision is within the control of the gene regulation built into the development model. Certain protein thresholds may also be explicitly declared. Explicit control of the ordering of requested actions of proteins - grow, change type, transcribe etc, or ordering of such requests at the cell level, will provide further control to gene regulation and its effects. Such control mechanisms are often included because of the sequential nature of the implementation medium as opposed to the parallel nature of the process modelled. Further, the proteins themselves are likely to have some form of explicit pre-condition to be met before the protein will be transcribed. It should be noted that most models using such a form of gene to proteins interpretation, miss out the mRNA step.

Instead of presenting the DNA as genes transcribing to proteins, some development models are presented more in terms of rules, L-systems [25] type approaches. Similar to the protein approach, such rules have pre-conditions that need to be met for the rule to be activated. However, most commonly, the pre-condition involves the neighbourhood types — functions. In essence, such models are in principle very similar to the DNA/protein models described, regulating at the level of the genes.

Another form of model is that of a development program. Such a form of development may be said to be more abstract than the two already presented. There is no specific DNA. Instead the cell program controls how inputs to the cell are interpreted so as to produce cell outputs. Such a model, may be said to have cell regulation rather than regulation at the level of individual genes.

DNA does not work alone but in multiple environments at different abstraction levels. Following the classification, given in [20], the DNA resides within a cell providing the inter-cell environment termed the *cell metabolism*. Models may provide for the notion of chemicals or proteins or other elements in the cell metabolism. Such elements provide some form of regulation, either by being included in pre-conditions to genes or they may control the effect of actions instigated by the DNA. This latter form provides more cell level regulation and is relevant to both DNA type models or cell program models. Such elements may also effect the next environment level - *inter-cell environment* which enables communication between neighbours. Information exchanged may include elements from the cell metabolism - proteins or chemicals or cell level information as in type i.e. functional information about neighbouring cells. Whichever element is transferred, the goal is similar. Such information is applied to the regulation of a cell's genes through the pre-conditions or may be built into the cell program. A further environmental influence is termed the *initial environment*. This relates to the assumed initial state of a cell which provides an initial path direction, influencing the path of development i.e. the effect of regulatory decisions. Finally, there is the *external environment*, providing information from out with the organism itself and which may be included directly into the gene pre-conditions or at the cell level, so as to effect the actions resulting from the gene regulation.

Other abstraction levels exist: organs, tissues etc which may be applied to extend this terminology. Whichever terminology is applied, a hierarchical structure exists within the cells environment, with different levels of control of gene regulation. In addition, further regulation is often included in a model in an attempt to counteract the challenge of implementing such a parallel process into a sequential medium.

III. Development Model

In the development model presented herein, an organism is bound by a three dimensional grid with a fixed size. A location inside the grid may be either empty or filled by a cell within the organism and a given cell has a type (function). 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 grid and the organism can develop up to the boundaries of the grid.

Inside each cell is a DNA, 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 from the set of allowable colours.

The number of development steps allowed and the bounds 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.

The genotype consists of a number of genes. Each gene has a promoter and coding region. The promotor contains the promotor ID (gene marker), 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 cell's 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. As such, the inter-cell environment in the form of type (colour) information directly effects the regulation of the genes in a cell. Further, since there is no feedback of the type(colour) of the cell itself, then the cell level does not directly effect this regulation.

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. The introduction of chemical thresholds to these chemicals provides further explicit gene regulatory control. Such chemicals provide the cell metabolism gene regulation in the model.

A proteins precondition is fulfilled if the state of the neighbours matches the neighbourhood pattern and if all the cell's chemical levels exceed 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. In this way, the proteins do not play an explicit role in gene regulation by the cell metabolism even though they reside in the cell metabolism as in the chemicals. However, they play an implicit role, after the more explicit ordering of events at the cell level.

The time to live specifies how many development steps the protein is active before it dies and is removed from the cell. This factor has a direct effect on the role of the protein in the gene regulation. In the extreme case, if there are no proteins in the cell metabolism, no genes will be transcribed and development will stop.

A. 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. Then, an organism is created using the given bounds and the cell is placed in the middle of this organism. The development process is now initialised.

The process of development is divided into discrete time steps — development steps. At a given step (clock tick) the same series of discrete events occur: the chemical concentrations are adjusted; the proteins are notified that a tick has occurred and the cell performs the actions requested by the proteins. As such, each tick 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, a practical, non biological event.

Event 2 the protein updates involves requesting the actions described in the functionality of those genes whose protein preconditions are met.

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 describes the process for a single cell. In nature the cell update across the organism is a highly parallel process. However, cell update itself 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 an empty neighbouring cell, this cell is then active and no other cell can grow into this cell. Both these conditions are regulatory decisions. The ordering of updates and the growth restrictions effect the path of development and thus the organism that may be developed.

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.

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 out with 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 processed. 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. The decision to allow growth in 6 directions at once and the introduction of stimulus, are all regulatory decisions at the cell level.

Changing the cell type has a similar stimulus process. However, a winner is chosen being that with the greatest stimulus level and if this level is over the threshold then the cell type is changed to the given type. Again this is a regulatory decision at the cell level.

The development process continues until the organism has developed for the given number of steps. The full-grown organism is then given as output of the system. It should be noted that development doesn't stop when the sought after shape is achieved. Instead it stops after the given number of steps and the sought shape may be achieved at any step. A feature of the model is that the organism will remain stabile once a perfect organism is achieved i.e. there is no further change in shape or colours.

IV. Protein Pre-conditions

As stated, the experiments presented herein do not provide the full picture of all regulatory effects within the developmental model presented. Instead two particular aspects are investigated — proteins and chemicals, which are explicitly specified in the pre-condition of the proteins. So what do these neighbourhood and chemicals represent in the given model?

The neighbourhood focuses on the neighbourhood of the cell i.e. the inter-cell environment. In the case of this model, the information exchanged with neighbours is that of type i.e. the functionality of each of the neighbours. By including such information in the pre-condition, the result of gene regulation in each of the neighbouring cells, at the previous point in time is being allowed to effect regulation of a gene — assuming discrete simulation.

When making such a decision, a further decision is needed. If this pre-condition is too specific i.e. states the exact functionality required for each neighbour, it may be hard to find a match for development to start or it may cause development to stop prematurely. Evolutionary techniques may be applied to refine the search but this does not prevent development stopping up. One way to introduce more freedom is to introduce don't cares into the neighbourhood pre-conditions, so as to provide more explorative freedom, enabling rules to fire or genes to be transcribed and to study the effects. Evolutionary techniques can then be applied to tune the specification of the neighbourhood as knowledge of the development of the organism is gained through fitness evaluation.

The chemicals in this particular model represent the cell metabolism. Distribution of chemicals is not present in the model and, as such, chemicals do not contribute to the intra-cell environment. Evolutionary techniques are not applied to such an environment, it is the transcription of genes, with actions of producing or consuming chemicals that effects the chemical concentration in the next time step. It should be noted that in addition to the thresholds themselves, other constraints are introduced in the model that effect their regulatory function. When the quantity of chemicals exceeds the max or min chemical bounds in a cell, these quantities are artificially fixed to the max/min values.

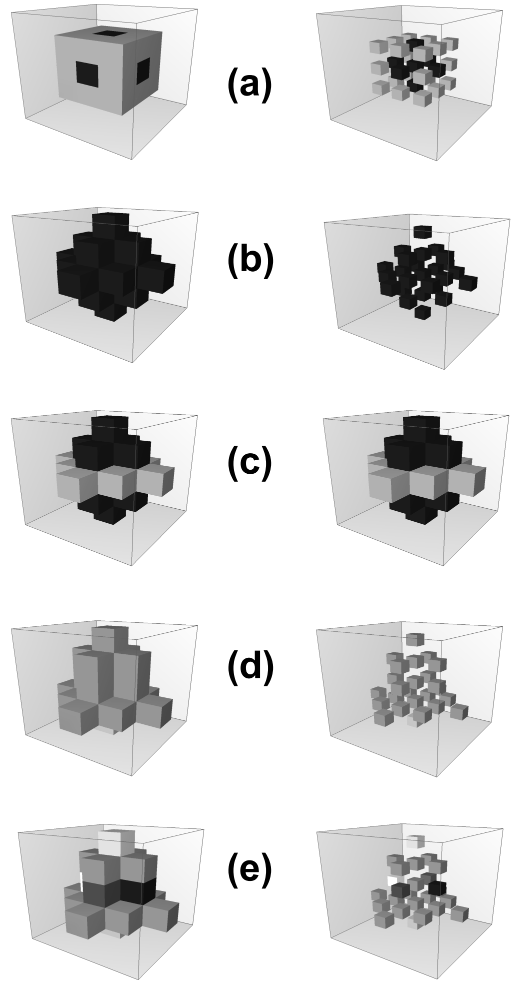

Fig. 1. Targets in Normal(left) and Exploded View(right), (a) cube with a cross, (b) tree, (c) Xmas Tree, (d) sphere and (e) divided sphere

V. Experiments

As stated, the goal of these experiments was to study the effect of two pre-conditions - chemicals and neighbouring types on gene regulation and on the organism achieved. The two measures investigated were the chemicals (CHT) and the Initial Don't-Care Neighbours (IDN). Chemicals were allowed to range from 0 to 5 and 10 chemicals. The number of chemicals is set for a given experiment and thus neither development nor evolution can change the number of chemicals. However, genes can transcribe to proteins which request changes in chemical concentrations. The neighbourhood of the cell consists of the types of its six neighbours and is a further pre-condition to the genes in the artificial DNA. From the discussion in section IV on specificity, it was important to be able to vary specificity and study the result with respect to the organism achieved and wrt different organisms. The Initial Don't care Neighbours (IDN)s relates to the number of neighbouring types, in a genes pre-condition, that are given as don't care in the initial DNA given to evolution and development. The IDNs were varied from 0 to 6. Unlike CHT, these are just initial values and, as such, change during evolution and development. In fact, it is evolution that refines the DNA to find a more fit DNA for the development process.

To be able to study the effect of one pre-condition, each pre-condition was studied in the light of the range of different parameters for the other pre-condition. Thus for experiments focusing on CHT, results plotted represent an average fitness based on the fitness achieved in the set of experiments representing all runs with a combination of this CHT value and any of the IDN values. Similarly for the IDN values, the average of all the experiments with all the CHT values was calculated. In each case, standard deviations were calculated. The experiment set resulted in 980 experiments for each of the shapes studied (illustrated in Figure 1).

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. 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

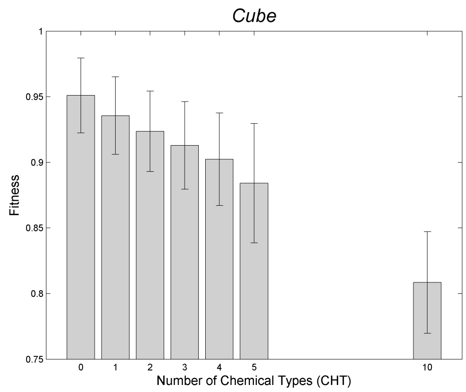

Fig. 2. Average Fitness for the Cube with varying CHT

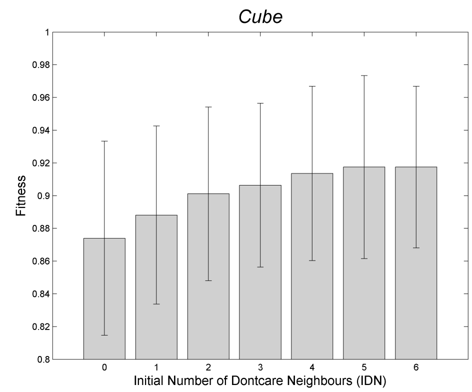

Fig. 3. Average Fitness for the Cube with varying IDN

incorrect place then no fitness credit is given. By rewarding shape, even when the type is wrong, provides more detailed feedback to the evolutionary process than would be the case if fitness was only based on the cells being correct or not. Further information regarding the parameters of the genetic algorithm may be found in [19].

The development setup involved a 6 bit promotor, 5 initial genes in the DNA, 3 proteins types (dividecell,transcribe gene and changetype), 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.

A. Cube

As illustrated in figures 2 and 3, low CHT and higher IDN seem to be the key to achieving high fitness. Further, the standard deviation remains relatively constant with increasing CHT or with decreasing IDN.

The fact that an increasing number of chemicals may have a negative effect in such a model is due to the fact that as you increase the number of chemicals - the specificity of the pre-condition increases. Each chemical has to reach a given threshold and all chemicals need to have reached the given threshold specified such that a match is achieved. As specificity increases, the chance of ending up in a local optima increases.

The standard deviation in the IDNs is fairly even, illustrating that there is a relative stability in the results achieved and no particular IDN level seems to lend itself especially to chance results.

B. Tree

Similar experiments were conducted for the tree, as illustrated in figures 4 and 5. Much of the same trend can be seen here: higher fitness is achieved with low CHT and high IDN. What is interesting here is the variation in the standard deviation.

Looking at the chemical case, it is the high chemicals that stand out with a much smaller deviation. However, it is likely that in the case of 10 chemicals, the specificity is so large that it is unlikely that a more fit individual is achieved by chance.

For the case of the IDNs much variation in the standard deviation may be seen. It would seem that as the IDNs go from 0 to 3, the standard deviation is increasing, perhaps indicating that there is more chance of hitting a fit individual by chance as specificty decreases. However, why is this not the case with the Cube? This has perhaps something to do with the complexity of the problem. Although the tree structurally is perhaps not so much more challenging, the challenge is in the types. In this model, neighbours are communicating type information and evolution is tuning such type information in the pre-condition. However no 100% fit individual was found. The best individual had one type error and that was in the colour of the base of the tree which should have been orange rather than green. Why is this the case? It is likely to be an EA challenge rather than development itself.

An EA is being applied to tune the pre-condition of the proteins. As discussed earlier, evolution seems to be able to tune development more appropriately to the needs of the application as the IDN increases. That is development is given a more tuned pre-condition to the needs of the application as the generations continue. Unlike the sphere, as the number of IDNs increase, the deviation increases. Due to the difficulty of the problem, chance plays a larger role and increasing IDNs provides more chance of hitting the correct development path. However, as the results show deviation slows up at the highest IDNs. It is, therefore assumed that maybe too much randomness is introduced to this difficult problem, reducing the chance of finding such chance solutions. Further, to achieve higher fitness the challenge of the error type needs to be met. The problem being that if fitness is tuned to the needs of the shape and the majority of the types i.e. green, then the more seldom type change gene is likely to be evolved away, providing development no chance of developing cells of such a type.

C. Christmas Tree

Very similar results, including deviation patterns were seen for the christmas tree. However, here the challenge is slightly

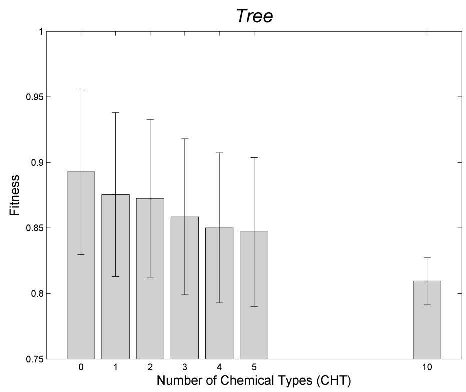

Fig. 4. Average Fitness for the Tree with varying CHT

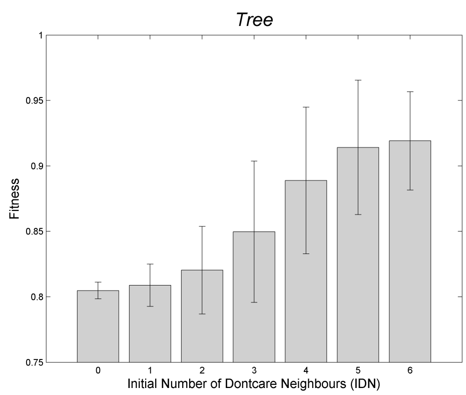

Fig. 5. Average Fitness for the Tree with varying IDN

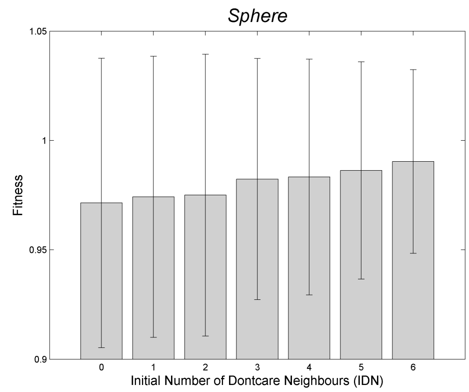

Fig. 6. Average Fitness for the Sphere with varying IDN

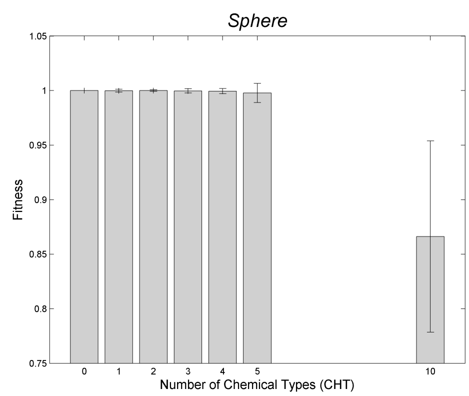

Fig. 7. Average Fitness for the Sphere with varying CHT

different. There is not just one type change to evolve away but many - all the decorations on the tree as well as the foot. In this case, it seems that the development model is prioritising shape first and then specialisation and, in particular, the achievement of the correct green cells. The non green cells were all correct in that they were non green but were blue - the original colour of these cells. Looking to fitness a clue may be seen.

As stated, fitness is allocated as 2 for cell in place and right colour; 1 for cell in place and wrong colour and 0 for cell not in place. As such, fitness is prioritising structure over type. However, this is also a logical way to apply fitness as one cannot have the correct type without a cell in place first. This prioritising of structure may cause a more likelihood of type information i.e. the appropriate genes to be evolved away. Similar to the tree instance, the type change from blue to green is the dominant type change and thus least likely to be evolved away.

D. Sphere

In the case of the sphere, perfect fitness was achieved in 856 of 980 experiments, indicating that the shape is relatively simple to achieve for evolution and development. Further, in this case the number of IDNs and CHT play a much less significant role, as shown in figures 6 and 7 respectively. For chemicals, apart from the extreme case of 10 chemicals, very little effect of increasing CHT can be seen and only a slight deviation change. However, for the IDNs the deviations indicate quite a high chance of finding a correct solution at random due to the simplicity of the problem.

E. Divided Sphere

In the divided Sphere, a more marked tendency to the benefit of high IDN and low CHT can be seen in figures 9 and 8 respectively. Comparing this with the results from the sphere, we see that this tendency has to mainly be due to the introduction of differentiation thus confirming our suspicions in the tree and christmas tree experiments. Studying the standard deviations of the Sphere, one can see that in the case of 10 chemicals, although poor fitness is achieved as is now expected, the standard deviation is not as small as other similar experiments. This is likely to be due to the simplicity of the shape itself. Even with the difficulty introduced with such a chemical specificity, a chance result is still possible with such a simple shape.

F. Summary

The results illustrate that choices made in the pre-conditions have quite an effect on the developing organisms and, for the

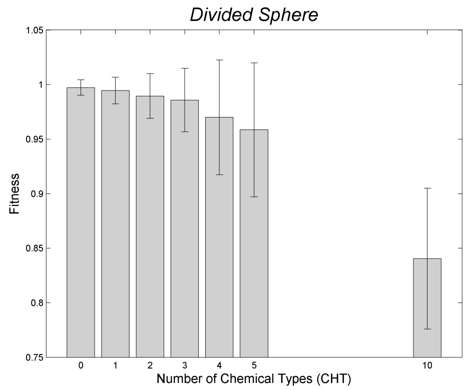

Fig. 8. Average Fitness for the Divided Sphere with varying CHT

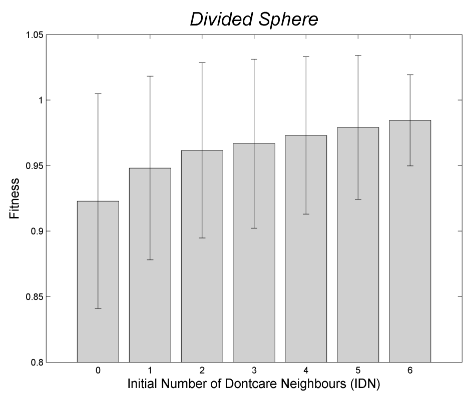

Fig. 9. Average Fitness for the divided Sphere with varying IDN

case of this particular model developing 3D shapes, a focus on few chemicals and higher IDN is required.

The standard deviations in each of the results highlighted much useful information. In some cases, although the challenge of achieving a fit individual increased with decreasing IDNs or increasing CHT, standard deviations showed a relative stability in the approach. However, in general, the extreme case of 10 chemicals or developing more difficult shapes produced less stabile results. These results show that even with the introduction of low chemicals and higher IDN, chance may still play a significant role.

In general, most experiments reached their optimal fitness after rather few development steps i.e. within the boundary of the length of development allowed. As such, it is not development time that is limiting better fitness but development is stopping. The results indicate that, due to the fitness calculation in the model, evolution has a tendency to evolve away less prioritised types, leaving development less chance of developing 100% fit individuals.One possible solution might be to have some form of weighted fitness to ensure that some weight is always given to the type. However, this will need some further investigation.

Assuming appropriate IDNs and CHT are applied for the organism (application) sought, such a regulatory approach can be expected to produce good phenotypes, on average.

Further, the results highlighted herein suggest that further investigation into the other regulatory effects within this model are needed, together with a refined fitness evaluation so as to define more of the rules for a development model for 3D shapes.

VI. Conclusion

The work herein, has investigated regulatory decisions in a development model in the light of DNA style and cell program style models. In particular, the work has drawn attention to regulation at different environmental abstraction levels. The regulatory decisions made in a case study DNA style development model for 3D shapes have been discussed.

Results of experiments towards a range of shapes provided a clear trend for setting the parameters of the pre-conditions i.e. low CHT and relatively high IDN. That is a rule has been found for the setting of these parameters for 2D and 3D shapes of limited complexity. Further, results indicate that as the complexity of the shape increases, although the same trend may be seen, much more deviation in the results is to be expected. A better fitness measure may help but further investigation into the regulatory mechanisms of the model is needed.

Looking at these results at a more general level, the IDN experiments investigate specificity in the precondition with respect to information from the inter-cell environment. It may be assumed, therefore, that whether neighbours exchange type or other information, higher IDNs may be advantageous. As these are initial values, their effect is likely to be on the initial stages of development. The results may indicate the importance of having the freedom to find the right path of development from the start so as to have any hope of achieving the goals sought. However, more investigation is needed.

CHT reflects information from within the cell i.e. from the cell metabolism. Again, more generally, less specificity with respect to the amount of information needed to match to fire a gene, is an advantage. In fact in this case no CHT produces the best results i.e. no exchange of information from the cell metabolism is advantageous. This is surprising as the cell metabolism has an important role to play in nature.

The chance solutions indicated in the results are also important with respect to generalisation. How many developed organisms in the literature are, in fact, found by chance. For the establishment of development as a technique, a more stabile method is sought through the achievement of a better understanding of gene-regulation in our developmental models. The work herein has provided some initial guidelines with respect to protein pre-conditions and their regulation by their environment but much work remains before any clear rules can be found.

References

[1] G. M. Rubin and et al, "Comparative genomics of the eukaryotes," in *Science*, 2000, pp. 2204–2215.

[2] J. Bongard and R. Pfeifer, “Evolving complete agents using artificial ontogeny,” in *Morpho-functional Machines: The New Species (Designing Embodied Intelligence)*, F.Hara and R. Pfeifer, Eds. Springer-Verlag, 2003, pp. 237–258.

[3] D. Federici, “Evolving complete agents using artificial ontogeny,” in *Proceedings of SAB, Simulation of Adaptive Behavior*, 2004, pp. 373–384.

[4] P. Haddow, G. Tufte, and P. van Remortel, “Shrinking the genotype: L-systems for ehw?” in *Proceedings of The 4th International Conference on Evolvable Systems: From Biology to Hardware, ICES2001*, ser. Lecture notes in computer science, vol. 2210. Springer-Verlag, 2001, pp. 128–129.

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

[6] G. Tufte and P. C. Haddow, “Building knowledge into development rules for circuit design,” in *Proceedings of 5th International Conference on Evolvable Systems: From Biology to Hardware, ICES2003*, ser. Lecture notes in computer science, A. M. Tyrrell, P. C. Haddow, and J. Torresen, Eds., vol. 2606. Springer-Verlag, 2003, pp. 117–128.

[7] A. Tyrell and et al, “Poetic tissue: An integrated architecture for bio-inspired hardware,” in *Proceedings of the 5th International Conference on Evolvable Systems: From Biology to Hardware, ICES2003*, ser. Lecture Notes in Computer Science, vol. 2606/2003. Springer Berlin / Heidelberg, 2003, pp. 269–294.

[8] H.Liu, J. Miller, and A. Tyrrell, “An intrinsic robust transient fault-tolerant developmental model for digital systems,” in *WORLDS Workshop at GECCO04*, 2004.

[9] L. Sekanina and M. Bidlo, “Evolutionary design of arbitrarily large sorting networks using development,” in *Genetic Programming and Evolvable Machines*. Springer Netherlands, 2005, vol. 6, no. 3, pp. 319–347.

[10] F. Gruau and D. Whitley, “Adding learning to the cellular development of neural networks: Evolution and the baldwin effect,” *Evolutionary Computation*, vol. 1, no. 3, pp. 213–233, 1993.

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

[12] P. Eggenberger, “Evolving morphologies of simulated 3d organisms based on differential expression,” 1997. [Online]. Available: citeseer.ist.psu.edu/article/eggenberger97evolving.html

[13] P. E. Hotz, “Comparing direct and developmental encoding schemes in artificial evolution,” in *IEEE Congress on Evolutionary Computation*. IEEE Computer Society, 2004.

[14] P. Hogeweg, “Evolving mechanisms of morphogenesis: on the interplay between different adhesion and cell differentiation,” in *Journal of Theoretical Biology*, vol. 203, 2000, pp. 317–333.

[15] 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*, vol. 1. Morgan Kaufmann, 1999, pp. 35–43.

[16] F. Dellaert, “Toward A biologically defensible model of development,” Master’s thesis, Case Western Reserve University, nov 1995.

[17] 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.

[18] J. F. Miller, “Evolving a self-repairing, self-regulating, french flag organism,” in *Proceedings of the Genetic and Evolutionary Computation Conference*, 2004, pp. 129–139.

[19] P. C. Haddow and J. Hoye, “Achieving a simple development model for 3d shapes: are chemicals necessary,” in *GECCO07*, 2007, pp. 1013–1020.

[20] G. Tufte and P. C. Haddow, “Achieving environmental tolerance through the initiation and exploitation of external information,” in *IEEE Congress on Evolutionary Computation*, 2007, pp. 2485–2492.

[21] P. K. Lehre and P. C. Haddow, “Development mappings and phenotypic complexity,” in *IEEE Congress on Evolutionary Computation*, 2003, pp. 62–68.

[22] D. Roggen and D. Federici, “Multi-cellular development: Is there scalability and robustness to gain?” in *Parallel Problem Solving from Nature - PPSN*, 2004, pp. 391–400.

[23] T. G. Gordon and P. J. Bentley, “Development brings scalability to hardware evolution,” in *NASA/DoD International Conference on Evolvable Hardware*. IEEE Computer Society, 2005, pp. 272–279.

[24] G. Tufte, “Phenotypic, development and computational resources: Scaling in artificial development,” in *GECCO08*, 2008.

[25] A. Lindenmayer, “Mathematical models for cellular interactions in development,” in *Journal of Theoretical Biology*, 1968.