Proceedings of the 2007 IEEE Symposium on Computational Intelligence in Bioinformatics and Computational Biology (CIBCB 2007)

Co-option and Irreducibility in Regulatory Networks for Cellular Pattern Development

Ranjitha A. Dhanasekaran
Department of Computer Science
Utah State University, Logan, USA
Phone: (435) 797-2432, email: ranjitha@cc.usu.edu

Gregory J. Podgorski,
Department of Biology
Utah State University, Logan, USA
Phone: (435) 797-3712, email: podgorski@biology.usu.edu

Nicholas S. Flann
Department of Computer Science
Utah State University, Logan, USA
Phone: (435) 797-2432, email: nick.flann@usu.edu

Abstract **– *We used a computational approach to examine three questions at the intersection of developmental biology and evolution: 1) What is the space available for evolutionary exploration for genetic regulatory networks (GRNs) able to solve developmental patterning problems? 2) If different GRNs exist that can solve a particular pattern, are there differences between them that might lead to the selection of one over another? 3) What are the possibilities for co-opting GRN subcircuits or even entire GRNs evolved to solve one pattern for use in the solution of another pattern? We used a Monte Carlo strategy to search for simulated GRNs composed of nodes (proteins) and edges (regulatory interactions between proteins) capable of solving one of three striped cellular patterning problems. These GRNs were subjected to a knockout procedure akin to gene knock-outs in genetic research. Knockout was continued until all network components of the reduced GRN were shown to be essential for function. This GRN was termed irreducible. We found many different unique irreducible GRNs that were able to solve each patterning problem. Thus, the space for evolutionary exploration for pattern-forming GRNs is large. Irreducible GRNs that solve a particular pattern differed widely in their robustness—the ability to solve a target pattern under different initial conditions. These differences may offer a target for selection to winnow out less robust GRNs from the set of unique GRNs. Finally, subgraph isomorphism analysis revealed great potential for co-option during evolution. Some irreducible GRNs appear in their entirety within larger GRNs that solve different patterning problems. In this case, the sub-GRN is a module. At much higher frequency, subcycles are shared widely among irreducible GRNs, including those that solve different patterns.***

Keywords **– *Genetic regulatory network, GRN, subgraph isomorphism, co-option, evolutionary dynamics, modularity, development, pattern formation, subcircuit, self-organization.***

I. INTRODUCTION

Genetic regulatory networks (GRNs) that control development are being deciphered through experimental approaches propelled by advances in genomics and systems biology [3]. These GRNs for development are complex and robust, generating reproducible outputs over a broad range of initial conditions [5]. Common themes in network architecture and regulatory logic are beginning to emerge (see [4]; [2]; [3]), including the use of evolutionarily conserved regulatory modules (kernels) and the existence of smaller regulatory circuits that have been repeatedly co-opted for different ends in development [3].

Since evolution is not a directed process, existing GRNs may not be the only ones capable of controlling a particular aspect of development. Instead, GRNs in nature may represent the one solution that was stumbled upon in the evolutionary history of a lineage. Once the solution was "discovered," it may have been frozen in place. The high degree of conservation of many GRN architectural elements in animal development suggests this possibility—but only if there are other solutions open for random evolutionary processes to discover.

We are interested in learning the dimensions of the developmental GRN solution space for evolutionary exploration. We approached this question by searching for GRNs that would solve three different patterns that resemble the striped patterns of pair-rule gene expression seen in the development of *Drosophila* and other insects (see Fig. 1). Building on our earlier work [10], we used a Monte Carlo strategy to identify networks that solve the three striped patterns. A genetic knockout algorithm, similar in logic to the use of targeted gene knockouts to understand biological GRNs, was employed to remove network components one-by-one until an irreducible network capable of solving the pattern was discovered.

Our results suggest that the solution space for evolutionary exploration is large and highly interrelated. They also show that although many different irreducible GRNs can solve a specific developmental patterning problem, there are significant differences in the robustness of the networks. These robustness differences may generate a strong selective pressure capable of winnowing out less robust GRNs. Finally, we show that there are many possibilities for evolutionary co-option of GRN subcircuits or even entire GRNs to solve new patterning

problems.

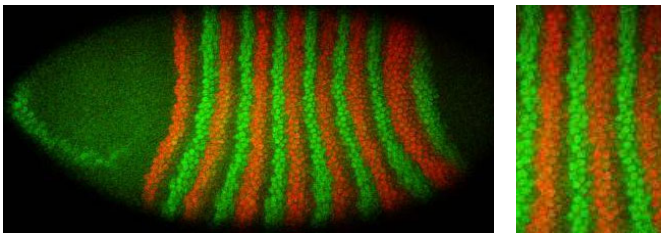


Fig. 1. A striped gene expression patterns in early *Drosophila* development. Odd-skipped (green) and even-skipped (red) expression domains at cleavage cycle 14. Left panel shows the nearly complete embryo; right panel is a closer view where each point is a cell nucleus. Image courtesy of Dr. J. Reinitz. Mt. Sinai School of Medicine.

II. APPROACH

GRNs can be described as a graph, where each node represents a protein's expression level and each edge represents a regulatory interaction between proteins. A protein is influenced when its production or degradation is controlled as a function of another protein's expression levels. In our model, the activity of a protein is directly related to its concentration. Since production and degradation are defined as rates of change, the GRN is naturally modeled as a set of coupled differential equations. Fig. 2(a) shows an example of a 3 protein, 4 edge GRN represented as a graph and Fig. 2(f) shows the same GRN as a set of coupled differential equations.

Table I illustrates the edges that represent protein interactions considered in this study. An in-edge j for protein P_0 contributes to the rate of change of P_0 as a weighted expression of one or two other proteins P_1, P_2 present in the same cell or in neighboring cells, where ω_j is the strength of the influence of j ($0.0 \leq \omega_j \leq 1.0$). Limiting functions $f(x) = \frac{x^2}{(1+x^2)}$, $g(x) = 1 - f(x)$, and $h(x) = \frac{2}{(1+e^{-x})} - 1$ are employed to model saturation effects in protein production and degradation.

Within an individual cell, protein expression can be controlled by a single protein (the direct control edges *C*-*F*, *H*, *I*) or some function of multiple proteins (the combinatorial control edge *G*). Over the sheet of cells, proteins influence each other through both long-range and short-range signaling. Edges *A* and *B* implement long-range signaling through diffusion under different boundary conditions. Edges *J*-*P* implement short-range signaling, where a cell can sense protein expression levels in directly neighboring cells across contacting membranes as in [6]. Edges *J* and *K* signal with all neighbors using $n(\sigma)$, which returns the set of directly neighboring cells. Edges *M*-*P* enable a cell to signal to a specific geometric neighbor cell using n_S, n_W, n_N, n_E, which return the cell directly neighboring to the south, west, north and east respectively. Such directional signalling is used in the embryo to build internal segment borders [5] and relies on morphogenic gradients that establish anterior-posterior and dorsal-ventral axes.

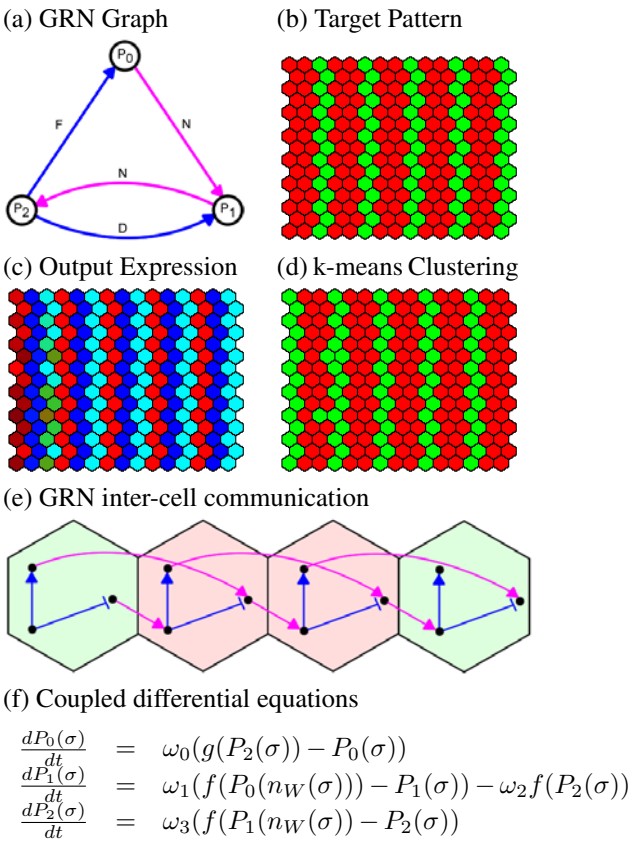


Fig. 2. Example of an irreducible GRN (a) found by our system for solving the *2-skip-1* target pattern (b). The letters on the edges of (a) indicate edge types shown in Table I. The output protein expression levels are shown in (c). Panel (d) shows the result of k-means clustering of the output expressions levels from (c). The GRN mapped onto a strip of cells (e) and the its coupled differential equations are also shown (f).

There are $d^e p^e p^{2p(e-p)}$ possible GRNs with p proteins and e edges ($e \geq p$ and d edge types). Fig. 2(a) shows an example GRN with 3 proteins and 4 edges discovered by the Knockout search method described in the following section. To determine the coupled differential equations of a GRN, the equations of each edge are composed, as illustrated in Fig. 2(f). To solve a GRN implemented in a sheet of q cells, each cell's protein values are first set from a uniform random distribution $[0.0, 1.0]$, then the $p \times q$ differential equations are numerically solved using the the Runge-Kutta method with $dt = 0.05$ until a fixed point is achieved (where the average update error $\leq 10^{-8}$ per cell). Fig. 2(c) illustrates the output expression pattern formed when the differential equations in Fig. 2(f) are solved over a sheet of 15×15 cells. In this panel, the color of each cell is determined by mapping P_0 to the red level, P_1 to green and P_2 to blue.

In this work we study the space of GRNs that can solve a small set of striped patterning problems similar to the patterns observed in early *Drosophila* development. Fig. 1 shows an example from *Drosophila* development and Fig. 2(b) shows one of our model target patterns, referred to as a *2-skip-1*. Other patterns we studied are *1-skip-1* and *2-skip-2*.

TABLE I

THE POSSIBLE EDGES FOR A GRN

Label	Description	Definition
A	P_0 Diffusion with zero boundary conditions	$\frac{dP_0(\sigma)}{dt} = \omega_j \frac{\partial^2 P_1(\sigma)}{\partial x^2} = \nabla P_1(\sigma)$
B	P_0 Diffusion with fixed boundary conditions	$\frac{dP_0(\sigma)}{dt} = \omega_j \frac{\partial^2 P_1(\sigma)}{\partial x^2} = \nabla P_1(\sigma)$
C	P_0 direct expression by P_1	$\frac{dP_0(\sigma)}{dt} = \omega_j f(P_1(\sigma))$
D	P_0 direct degradation by P_1	$\frac{dP_0(\sigma)}{dt} = -\omega_j f(P_1(\sigma))$
E	P_0 driven to same as P_1	$\frac{dP_0(\sigma)}{dt} = \omega_j (f(P_1(\sigma)) - P_0(\sigma))$
F	P_0 driven to opposite of P_1	$\frac{dP_0(\sigma)}{dt} = \omega_j (g(P_1(\sigma)) - P_0(\sigma))$
G	P_0 driven to difference in values between P_1 and P_2	$\frac{dP_0(\sigma)}{dt} = \omega_j (h(P_1(\sigma) - P_2(\sigma)) - P_0(\sigma))$
H	P_0 autocatalysis and reciprocal control by P_1	$\frac{dP_0(\sigma)}{dt} = \omega_j (f(\frac{P_0(\sigma)^2}{P_1(\sigma)} + \psi_j))$
I	P_0 quadratic degradation by P_1	$\frac{dP_0(\sigma)}{dt} = \omega_j (g(P_1(\sigma)^2) - \psi_j)$
J	P_0 driven to same as cell neighbors values of P_1	$\frac{dP_0(\sigma)}{dt} = \omega_j (f(\sum^{\rho \in n(\sigma)} \frac{P_1(\rho)}{6}) - P_0(\sigma))$
K	P_0 driven to opposite of cell neighbors values of P_1	$\frac{dP_0(\sigma)}{dt} = \omega_j (g(\sum^{\rho \in n(\sigma)} \frac{P_1(\rho)}{6}) - P_0(\sigma))$
L	P_0 driven to difference in opposing cell neighbor values of P_1	$\frac{dP_0(\sigma)}{dt} = \omega_j (f(\sum^{\rho \in n(\sigma)} \frac{P_1(\rho) - P_1(op(\sigma,\rho))}{6}) - P_0(\sigma))$
M:P	P_0 driven to same as geometric neighbor value of P_1; with $i \in N, W, S, E$	$\frac{dP_0(\sigma)}{dt} = \omega_j (f(P_1(n_i(\sigma))) - P_0(\sigma))$

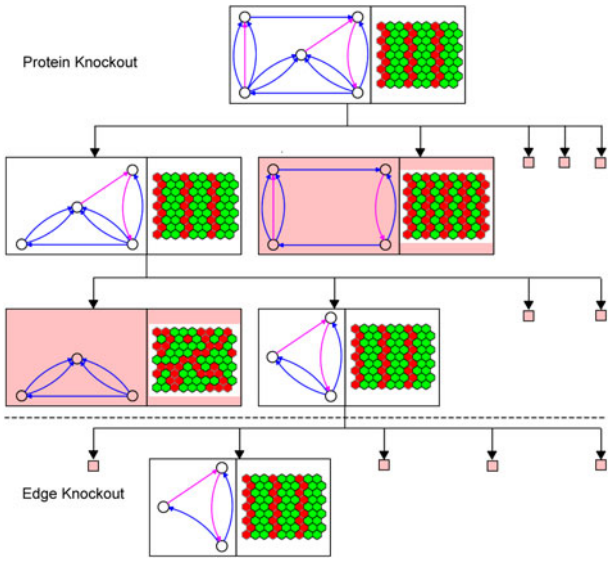


Fig. 3. An example tree created by running the Knockout method show in Table II on a *2-Skip-1* GRN discovered by Monte Carlo Search. Each GRN explored is shown with an example expression pattern. Nodes with light red backgrounds represent failure, where the pattern error exceeds the 10% threshold.

To quantify how well a GRN solves a particular target pattern, the protein expression levels that result from solving the coupled differential equations must be mapped to distinct cell types using a combinatorial code. Such codes partition the p-dimensional protein spectrum into distinct regions, each corresponding to a cell type [11]. In this work, k-means clustering is used to identify the combinatorial code which best partitions the cells into k-types. An example is given in Fig. 2(d), with $k = 2$ since there are two types in the target pattern. Two clusters are found defining the combinatorial code: type 1 (red) is P_0 high and P_1 low, and type two (green) is P_0 low and P_1 high. Each cell's assigned type is compared with the type assigned in the target pattern and a tally is made of misplaced cells. If the target pattern has alternative rotations (or phases), all rotations are tried and the best match is used as the error. Comparing Fig. 2(b) with Fig. 2(d) gives an error of 2 cells or 0.89%. To give an accurate evaluation, the GRN is solved 5 times under different random initial protein conditions and the mean error returned.

A. *Irreducible GRNs*

An irreducible GRN is one where the removal of any one component (protein or edge) results in loss of function. In this study, this is the inability of the GRN to accurately produce the target pattern. To identify irreducible GRNs a Knockout procedure, similar to those used in a genetic approach to the analysis of biological GRNs, was employed to reduce GRNs found by a Monte Carlo search of the GRN space (see [10] for details). This procedure starts with a GRN that accurately solves one of the target striped patterns and identifies the smallest possible sub-graph of the original GRN that still adequately solves that same target pattern.

Consider the knockout tree in Fig. 3. The parent GRN has 5 nodes and 12 edges representing the proteins and the interactions among them respectively. The knockout procedure is a two step process. First, the proteins are removed (knocked-out) one-by-one, then the edges are knocked-out one-by-one. At every node in the knockout tree, every component of the parent GRN is knocked-out one by one, and each resulting child GRN is measured for its fitness with respect to the target pattern. If a child GRN's error is less than or equal to the threshold value, then it becomes the parent GRN and is submitted to another round of the knockout procedure. The threshold value is the acceptable error that produces imperfect but clearly recognizable target patterns. In this study, the threshold was set to 10% misplaced cells. Only the first successful child GRN is pursued for further knockout because the order of deletion of components is irrelevant. Knockouts are repeated until none of the child GRNs meets the threshold criteria. In this case the parent GRN is considered to be the irreducible GRN. The irreducible GRN in this example has 3 proteins and 4 edges. During the knockout of this GRN, which produces a *2-skip-1* pattern, we noted that one of the failure GRNs accurately produced a *1-skip-1* pattern. This indicates a close relationship among GRNs for similar stripped patterns.

The knockout algorithm can be formally defined as a form of greedy depth-first search algorithm and is given in Table II. Two sub-routines are called: Solve(P, E) which initializes the proteins and then uses the Runge-Kutta method until a fixed point, and Error(T, S) which matches the target pattern T against the clustered protein expression pattern. The input to this algorithm is a GRN found by the Monte Carlo search to solve a particular pattern. The algorithm terminates when the

TABLE II
ALGORITHM DEFINING THE KNOCKOUT PROCEDURE

Algorithm Knockout(P, E, T)
Input: $P = \{p_1, p_2, \ldots, p_k\}$, GRN proteins
$E = \{e_1, e_2, \ldots, e_q\}$, GRN edges
$T =$ Target cell pattern
Output: $(\widetilde{P}, \widetilde{E}), \widetilde{P} \subseteq P, \widetilde{E} \subseteq E$ (a Knocked-Out GRN)
Begin
//Protein Knockout
for each $p_i \in P$
$P' \leftarrow P - \{p_i\}$
$\widehat{E} \leftarrow \{e_j \in E | e_j$ connects to $p_i\}$
$E' \leftarrow E - \widehat{E}$
if Error$(T$,Solve$(P', E')) \leq 10\%$
then return Knockout(P', E', T)
//Edge Knockout
for each $e_i \in E$
$E' \leftarrow E - \{e_i\}$
$\widehat{P} \leftarrow \{p_j \in P | p_j$ is isolated w.r.t $E'\}$
$P' \leftarrow P - \widehat{P}$
if Error$(T$,Solve$(P', E')) \leq 10\%$
then return Knockout(P', E', T)
return (P, E)
End.

attempt to delete each protein and each edge fails to produce the solution pattern. The output is an irreducible GRN that solves the target pattern T.

B. Co-option among Irreducible GRNs

Co-option is the use of an element of one GRN by another GRN, often for a distinct function. This work considers two kinds of co-optable network elements: modules and sub-circuits. We define a module as a fully functional GRN that appears as part of another larger GRN. A sub-circuit is defined as a GRN cycle that occurs in two or more GRNs. In contrast to modules, sub-circuits may or may not function as independent GRNs.

B.1 Modules: Subgraph Isomorphic GRNs

Modules are discovered by computing subgraph isomorphisms between all pairs of GRN graphs. A graph G_1 is a subgraph of G_2 if under some one-to-one mapping between the nodes of G_1 and a subset of the nodes of G_2, the in and out edges of all nodes in G_1 are always a subset of those of G_2. This problem is known to be P-space Complete in general. In this study, we are interested in determining subgraph isomorphism between directed graphs with distinct edge labels, which is known to be NP-Complete. The relatively small size of the irreducible GRN graphs (see Fig. 6 for some examples) makes it feasible to use an exact algorithm, whose performance was greatly improved through the use of edge-hashing [9]. All irreducible GRN graphs were compared with all other GRNs for all three patterns.

B.2 Subcircuits: Common Sub-cycles between GRNs

Subcircuits are computed by extracting all cycles from the GRN graphs and performing pairwise comparisons over all cycles. A cycle in a GRN graph of p proteins and e edges is a feedback loop comprising of between 2 and e edges and is characterized by the order of the edge types involved. To determine common cycles between graph G_1 and G_2, we first extract all cycles from each graph, then identify an intersection under all possible rotations of each cycle. The complexity of this computation is $O(p^2 e^3)$, which is feasible for the small GRN graphs studied here.

III. RESULTS

This computational study first discovered a distinct set of irreducible GRNs for each of the target pattern and considered their distribution with respect to size (number of nodes and edges). Next, the robustness of the irreducible GRNs was measured and analyzed with respect to network architecture. Finally, the common subgraphs and sub-cycles were identified.

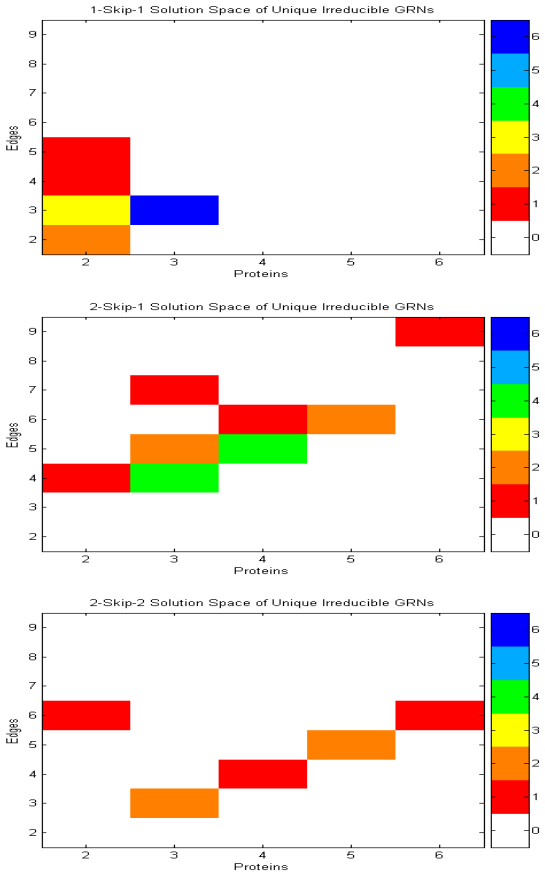


Fig. 4. The distribution of unique irreducible GRN's for the three target patterns. The scale maps each color to the count of GRNs.

A. Identification of Irreducible GRNs

An extensive Monte Carlo search of the GRN space was performed using 30 high performance workstations running in parallel for 10 days. This identified over 1000 GRNs with

errors below the 10% threshold over one of the three target patterns. A random subset of GRNs was selected for each target pattern to provide an approximately uniform distribution of GRNs over the space of network sizes $2 \leq p \leq 8$ and $p \leq e \leq 13$. Each GRN solution was then processed by the Knockout procedure shown in Table II to identify its irreducible GRN. Finally, isomorphic GRNs were eliminated from the set of irreducible GRNs using an edge hashing scheme [9]. This produced a total of 13 *1-skip-1* GRNs, 16 *2-skip-1* GRNs and 7 *2-skip-2* GRNs. The distribution of irreducible GRNs is shown in Fig. 4.

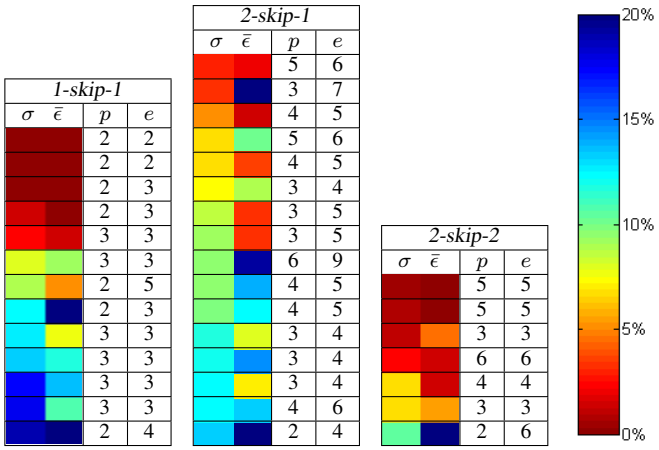


Fig. 5. The robustness of all identified irreducible GRN solutions for the 3 striped patterns. In each table, σ is the standard deviation of error, $\bar{\epsilon}$ is the average error, p is the number of proteins and e number of edges. The rows are sorted by σ so the most robust network is the at the top and the least robust network at the bottom. The key for the colors used to represent σ and $\bar{\epsilon}$ is on the right.

B. Robustness of Irreducible GRNs

Robustness measures the ability of a GRN to consistently produce a high quality pattern under varying initial concentrations of proteins and varying strengths of the interactions between them (ω_j in Table I). All the irreducible GRNs demonstrate robustness under varying initial concentrations.

To measure the robustness of the GRNs under varying edge strengths, we employed the same methodology used by von Dassow et. al [5] where they evaluated the robustness of a single network that creates segment polarity in *Drosophila*. The robustness of each irreducible GRN was determined by setting each edge strength ω_j, $(1 \leq j \leq e)$ in turn to a uniform random number in the range $0.1 \leq \omega_j \leq 1.0$ and then computing the error of the resulting pattern with respect to its target pattern. The process was repeated 40 times for each GRN, then the mean and standard deviation was computed. The results are illustrated in Fig. 5. The standard deviation of the pattern error quantifies the robustness of each individual pattern with low standard deviation implying high robustness. Examples of GRNs with high, median and low robustness for each target pattern are shown in Fig. 6.

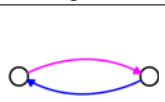
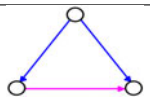
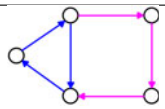
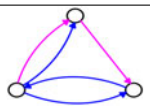
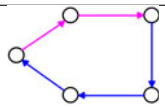
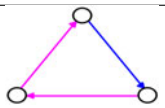
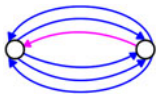
Pattern	Robustness		
	Highest	Median	Lowest
1-skip-1			
2-skip-1			
2-skip-2			

Fig. 6. Examples of irreducible GRNs discovered for the three target patterns. Edges are color coded based on whether they implement within-cell control (purple) or between cell signalling (red).

B.1 Modules: Subgraph Isomorphic GRNs

The results showing the sub-graph isomorphic relation between each pair of GRNs are given in Fig. 7. Each circle in the figure represents a unique GRN, with the clockwise order of each pattern corresponding to the order of robustness shown in Fig. 5. Our results show that modules among irreducible GRNs are rare. There are 630 possible uses of the GRNs as modules, but only 9 (1.4%) are employed. Significantly, there are no modules shared among GRNs that solve the same target pattern. This result supports the effectiveness of the knockout procedure for the discovery of irreducible GRNs.

There is a partial ordering relationship among the GRNs solving the three striped patterns. The *1-skip-1* GRNs can be used as modules within GRNs that solve the *2-skip-1* and *2-skip-2* patterns. The *2-skip-2* GRNs also appear as modules within the *2-skip-1* GRNs. We never detected a case in which a *2-skip-1* GRN was used as a module within the other pattern's GRN. This suggests that the *2-skip-1* pattern is more difficult to solve, perhaps because of its asymmetry.

B.2 Subcircuits: Common Sub-cycles between GRNs

The results showing the common sub-cycle relation between each pair of GRNs is shown in Fig. 8. A sub-cycle is feedback loop which occurs in two or more GRNs. A sub-cycle, which functions as a subcircuit in biological GRNs, may or may not operate as an independent pattern solving GRN. There are many more subcircuits among the GRNs than modules. Studies identified 56 (9%) out of a potential 630 pairs of GRNs that could share a subcircuit. An analysis of these 56 common subcircuits yields ten unique subcircuits, six 2-edge cycles and four 3-edge cycles. In contrast to modules, we found subcircuits that were shared between GRNs that solve the same pattern. The *2-skip-1* GRNs had 28 shared subcircuits, the *1-skip-1* GRNs had 14, while the *2-skip-2* GRNs shared none.

IV. DISCUSSION

Three questions drove this work: What is the space available for evolutionary exploration for GRNs able to solve developmental patterning problems? If different GRNs exist that can

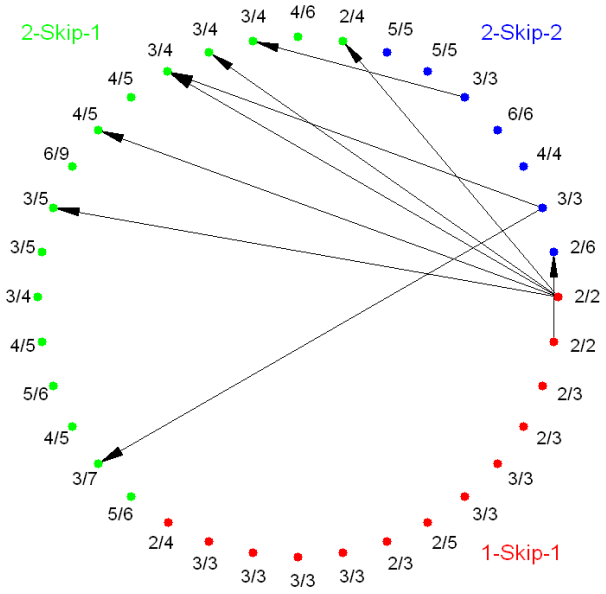


Fig. 7. Subgraph isomorphism (module) results. Each node in the circle represents a single irreducible GRN. Red is *1-skip-1*, green is *2-skip-1* and blue is *2-skip-2*. The node labels correspond to those in Fig. 5. An edge from GRN i to j means that i is a sub-graph of j.

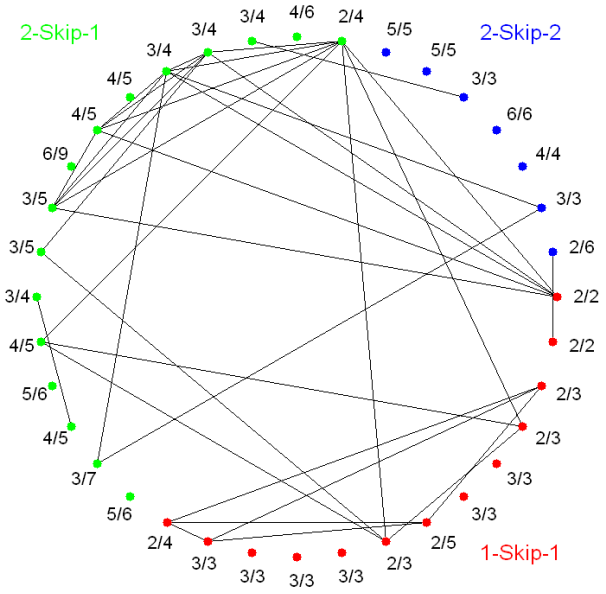


Fig. 8. Graph sub-cycle (subcircuit) results. Each node in the circle represents a single irreducible GRN. Red is *1-skip-1*, green is *2-skip-1* and blue is *2-skip-2*. The node labels correspond to those in Fig. 5. Two GRNs are connected if they share a common cycle.

solve a particular pattern, are there differences between them that might lead to the selection of one over another? What are the possibilities for co-opting GRN subcircuits or even entire GRNs evolved to solve one pattern for use in the solution of another pattern?

Our approach was to model GRNs composed of nodes (proteins) connected by a variety of edge types (regulatory interactions between proteins) that simulate many of the regulatory interactions that occur within and between embryonic cells. In our model, the same GRN that allows for intra- and intercellular interactions operates in every cell. Regulatory interactions can occur via diffusion, direct cell-cell contact signaling, or a combination of the two. Many of the interactions are directional, such that signaling occurs only in a "north-to-south" or an "east-to-west" direction. These directional interactions correspond well with the polarity of dorsal-ventral, anterior-posterior and medial-lateral signaling that occurs in biological development.

We searched for all GRNs within the range of 2 protein-2 edge to 8 protein-13 edge GRNs that could solve one of three cellular patterning problems. GRNs of a given complexity (number of proteins and number of edges) were subjected to a mutational knockout analysis similar to the approach of targeted gene disruption used in biological research to assess gene function. Proteins and the connections between them (edges) were removed to learn if they were extraneous or essential for GRN function. Knockout of extraneous components was continued until a GRN was discovered in which every component was essential for network function. This GRN was considered to be irreducible.

A. *The GRN Space Available for Evolutionary Exploration*

The space available for evolutionary exploration is the number of irreducible GRNs that can solve a given patterning problem. The evolutionary space is large for all three patterns we investigated. There were 16 unique, irreducible GRNs that solved the *1-skip-1* pattern, 13 for the *2-skip-1* pattern and 7 for the *2-skip-2* pattern (Fig. 5). The complexity of GRNs required to solve each pattern, the distribution of GRN complexity, and the density (number of unique, irreducible GRNs of the same number of proteins and edges) were different for the three patterns (Fig. 4). Some general trends included: 1) Lower network complexity was required for the *1-skip-1* pattern than for the other patterns; 2) There was a significantly broader distribution of network complexity for solutions of the *2-skip-1* and *2-skip-2* patterns; and 3) The evolutionary solution space of *2-skip-2* pattern was roughly half that of other patterns.

B. *Selection for Robustness*

Given that there are many possible GRNs open for discovery, are there differences between GRNs that would favor the ultimate selection of one? One feature of GRN operation that may provide a selective advantage is robustness in the face of varying initial conditions. We examined the robustness of each irreducible GRN over randomly generated and widely varying protein and edge-strength values. Large differences were seen in the robustness of GRN function (Fig. 5). For irreducible GRNs that solve *2-skip-1* and *2-skip-2* patterns, there is a trend for networks of intermediate complexity to be more robust than those at either extreme of complexity. In addition, it appears that GRN graphs with low degree and sparse cycles had higher

robustness. However, the most significant conclusion drawn from this part of the investigation is that although many GRNs are available for discovery, there are sharp differences between them. Only a small set of GRNs initially discovered by evolution is likely to remain after long-term selection.

C. There Are Many Opportunities for Co-Option

An important aspect of GRN architecture and its evolution is suggested from analysis of GRNs that control animal development. This feature is the existence of subcircuits and modules that appear frequently in GRNs employed in the development of divergent lineages, such as *Drosophila*, sea urchin, and the chordate *Ciona* [1]. The appearance of these simpler subelements opens the possibility for evolution by co-option of existing network modules. By combining previously evolved network features, new patterning problems can be solved.

We examined whether and to what extent the irreducible GRNs discovered here shared architectural features. Such shared features, especially if they appear frequently, may illuminate potential evolutionary trajectories through co-option in the evolution of pattern forming GRNs. We discovered extensive overlap in many architectural features in GRNs that solved the three different target patterns.

Network elements have been referred to by a variety of terms that include kernels, subcircuits, switches, and modules [3]. We searched for sharing of two types of network elements: modules and subcircuits. In this work, modules were defined as an element within a larger GRN that is capable of operating on its own to solve a pattern. A subcircuit is a conserved network feedback loop that may or may not operate as an independent GRN.

Four different modules exist within irreducible GRNs that solve the *2-skip-1* and *2-skip-2* patterns (Fig. 7). These modules are independent GRNs that solve the *1-skip-1* or *2-skip-2* patterns. In contrast, sharing of subcircuits is more extensive and complex (Fig. 8). Subcircuits are shared both within GRNs that solve a single patterning problem (e.g., see the extensive subcircuit sharing in many *2-skip-1* GRNs) and between GRNs that solve two different patterning problems (e.g., the extensive subcircuit sharing between the *1-skip-1* and *2-skip-1* GRNs.). In addition, some subcircuits that appear in the *1-skip-1* GRNs are used to solve more complex patterns. For example, a subcircuit within the 2-protein, 2 -edge *1-skip-1* GRN appears in 5 different GRNs able to solve the *2-skip-1* pattern.

The significance of these findings is that the possibilities for evolutionary co-option at the module and subcircuit levels are vast. Modules evolved for one purpose can be further evolved by the addition of a new protein or new interaction between existing proteins to form a new module for another purpose. Subcircuits utilized within one module can be duplicated and utilized by another module. Interestingly, subcircuits do not have to be independently functional to expand the potential of the evolutionary search space. The result of this co-option is often the creation of an expanded GRN capable of solving a completely different patterning problem. The GRN evolved by co-option is a mixture of the old and the new.

Although the focus of this work was on the evolutionary trajectories open for pattern formation during development, the conclusions concerning the wide space of GRNs available to evolution and the abundant opportunities for co-option to increase complexity are likely to apply equally well to other fundamental biological processes.

References

[1] E.H. Davidson (2006) *The Regulatory Genome: Gene Regulatory Networks in Development and Evolution*, First Edition, New York: Elsevier, 2006.

[2] M. Levine and E. H. Davidson, "Gene regulatory networks for development," *Proc. Natl. Acad. Sci.* U. S. A., vol. 102(14), pp. 4936-4942, 2005.

[3] E. H. Davidson and D. H. Erwin, "Gene Regulatory Networks and the Evolution of Animal Body Plans," *Science*, vol. 311(5762), pp. 796-800, February 2006.

[4] S. Istrail and E. H. Davidson, "Logic functions of the genomic cis-regulatory code," *Proc. Natl. Acad. Sci.* U. S. A., vol. 102(14), pp. 4954-4959, 2005.

[5] G. von Dassow, E. Meir, E. M. Munro, and G. M. Odell, "The segment polarity network is a robust developmental module," *Nature*, vol. 406(6792), pp. 188-192, July 2000.

[6] J. R. Collier, N. A. M. Monk, P. K. Maini, and J. H. Lewis, "Pattern Formation by Lateral Inhibition with Feedback: a Mathematical Model of Delta-Notch Intercellular Signalling," *J. theor. Biol.*, vol. 183, pp. 429-446, 1996.

[7] D. Roggen and D. Federici, "Multi-cellular Development: Is There Scalability and Robustness to Gain?" in *Lect. notes comput. sci.*, vol. 3242. Berlin: Springer, 2004, pp. 391-400.

[8] K. O. Stanley and R. Miikkulainen, "A Taxonomy for Artificial Embryogeny," *Artif. life*, vol. 9(2), pp. 93-130, 2003.

[9] P. C. Nguyen, T. Washio, K. Ohara and H. Motoda, "Using a Hash-Based Method for Apriori-Based Graph Mining," in Lect. notes comput. sci., vol. 3202. Berlin: Springer, 2004, pp. 349-361.

[10] N. Flann, J. Hu, M. Bansal, V. Patel and G. Podgorski, "Biological Development of Cell Patterns: Characterizing the Space of Cell Chemistry Genetic Regulatory Networks," in *Lect. notes comput. sci.*, vol. 3630. Berlin: Springer, 2005, pp. 57-66.

[11] A. Ghazi and K. VijayRaghavan, "Developmental biology:Control by combinatorial codes," *Nature*, vol. 408, pp. 419-420, November 2000.