Honda Research Institute EU
Honda Research Institute Europe GmbH
https://www.honda-ri.de/

Evolvability of Graph and Vector Field Embryogeny Representations

Till Steiner, Bernhard Sendhoff

2011

Preprint:

This is an accepted article published in IEEE Congress on Evolutionary Computation. The final authenticated version is available online at: https://doi.org/[DOI not available]

Evolvability of Graph- and Vector Field Embryogeny Representations

Till Steiner
CST AG
Darmstadt, Germany
Email: till.steiner@cst.com

Bernhard Sendhoff
Honda Research Institute Europe GmbH
Offenbach, Germany
Email: bs@honda-ri.de

***Abstract*—Most developmental representations for design optimization with evolutionary computation that have been described in the literature are graph-based mimicking the interactions observed in biological gene regulatory networks. Alternative methods that directly manipulate the dynamical control system for developmental processes have been termed Vector Field Embryogeny (VFE) and have been applied successfully to cell differentiation. In this paper, we compare the evolvability of graph-based and vector field representations for controlling developmental processes. Inspired by the notion of strong causality in evolutionary strategies, we measure the covariance between genotype and phenotype changes for both representations. Furthermore, we propose a measure to characterize the representational power of both methods. If we compare VFE and graph-based representations with similar representational power, we notice that the covariance measure and therefore, the expected evolvability of VFE is higher. We also observe that the representational power of both methods decreases with increasing degree of freedom. We speculate that the reason for this could be the increased probability of the occurrence of strong point attractors.**

I. Introduction

Artificial Development (AD) finds increasing interest in the scientific community of evolutionary computation, which is mainly due to its promise to achieve scalability in automated design [1]. AD systems can be characterized by the lack of a direct mapping from genotype to phenotype. Instead, these approaches use a simulated developmental process to translate genotypic information into the phenotype. This means that the dimensionality of the genotype space is decoupled from the complexity of the phenotype and in general can be smaller than the dimensionality of the phenotype space. By exploiting regularities in both the structure and the organization of the process, relatively short genotypes can result in complex phenotypes.

To achieve this structural "unfolding", usually an incremental, time dependent generative process is simulated (e.g., [2], [3], [4], [5], [6], [7], [8]). Due to this time dependency, a dynamical system is necessary to control the process. Several formal representations of such a dynamical system are conceivable. In this contribution, we will focus on two models for the dynamical control of development.

Most models that are used for the simulation of the dynamics of the developmental process in evolutionary computation are graph based. Of course, this is mainly due to the inspiration taken from biology where this process is governed by the interaction of genes in regulatory networks. As a representative graph formulation, we choose continuous time recurrent neural networks (CTRNNs). This representation has been frequently studied in the literature and it is conceptually very close to gene regulatory networks.

More recently, an alternative model for the developmental process has been suggested that uses a more direct phase space representation of the dynamics to describe complex systems. This model has been termed Vector Field Embryogeny and has been shown to offer some advantages over graph-based models both with respect to the explicit role of time [9] as well as to the evolutionary success on some simple cell differentiation tasks [10].

In general, the field of artificial development has not yet fulfilled the aforementioned high expectations with regard to solving the scalability problem in system design. One reason seems to be that simulated developmental processes are difficult to evolve and so far success has been limited to mostly simple systems. Therefore, it is our aim to analyze the evolvability of developmental representations exemplary for CTRNN and VFE, since they represent structurally different approaches to the problem of evolving dynamic control.

In general, an evolvable representation must enable the evolutionary algorithm to stabilize successful mutations and thereby to enhance the system quality with respect to the selection criteria step by step during the generations. In the context of evolutionary strategies this includes the capability to adapt the strategy parameters of the mutation distribution to the topology of the search space. One criterion to quantify the evolvability of a certain representation for evolutionary strategies is strong causality [11] and the measurement of correlations between changes in the genotype and phenotype space [12].

In more detail, our approach can be outlined as follows: firstly, we take a large sample of randomly generated genotypes. Secondly, we mutate each sample several times and monitor the effect of the mutations. Thirdly, we observe the resulting dynamics of the original and the mutated genotypes for both representations (CTRNN and VFE) individually. We define a phenotypic distance for dynamical control systems, and investigate the covariance between genotypic and phenotypic changes for each representation. Finally, we compare the

covariances in graph and VFE representations. Furthermore, we propose a measure of representational power, i.e., the capability to represent different dynamics in the light of developmental control. This will enable us to fairly compare both representations.

The paper is structured as follows: in Section II, we outline the two representations (CTRNN and VFE) that we analyze in this paper. Section III introduces *strong causality* as a prerequisite for the success of the optimization with an evolutionary strategy. Then, section IV defines a general phenotype for dynamical systems that can be employed to monitor the phenotypic effect of mutations. In Section V, the experiments and results are described for both representations separately and in Section VI our observations are discussed for both CTRNN and VFE. We conclude the paper in the final section.

II. Graph- and Vector Field Embryogeny-based dynamics representation

A. Artificial Neural Networks as Models for GRNs

Graph representations in artificial developmental systems are usually inspired by natural gene regulatory networks (GRNs), and control a multi-cellular development. For simplicity reasons, only single-cell systems are simulated in our analysis. Since the simulated cell has no neighbors, there is no need to simulate interactions e.g. by using a diffusion process. Note, that our approach could be easily extended to multi-cellular systems (see section VII). At the same time, in this contribution this simplification allows us to regard GRN models as pure graphs, which can be described by a connection matrix and nonlinear activation functions. The effect of diffusion in a single cell simulation could be incorporated by introducing a negative auto-regulation at each node, if the calculation domain can be expected to have large size or open boundary conditions. With these simplifications in mind, we replace the GRN formalism with the widely used structure of continuous time recurrent neural networks (CTRNNs). In these neural network models, the activation a_i of the i-th neuron is calculated by $a_i = \frac{1}{1+e^{-s}}$, where s is the weighted sum of the inputs to this neuron. Weights are in the range $[-1, 1]$.

B. Vector Field Embryogeny

Vector field embryogeny shifts the representational focus directly to the dynamics that a network would generate, i.e., to its phase space. Vector Field Embryogeny enables mutation operators to directly influence a system's phase space (direct manipulation), instead of doing so indirectly via graph manipulation. We will briefly outline the VFE framework, for a more detailed introduction the reader is referred to [10].

Vector Field Embryogeny is inspired by vector field editing. In computer graphics, the vector field editing method is used for the creation of texture alignments and extraction of analytical information about given graphical representations of vector fields [13], [14], [15]. Even though this method is extendable to D dimensions by applying the respective D-dimensional geometric operations, the following considerations are presented using a two-dimensional version of the system for clarity and visualization purposes.

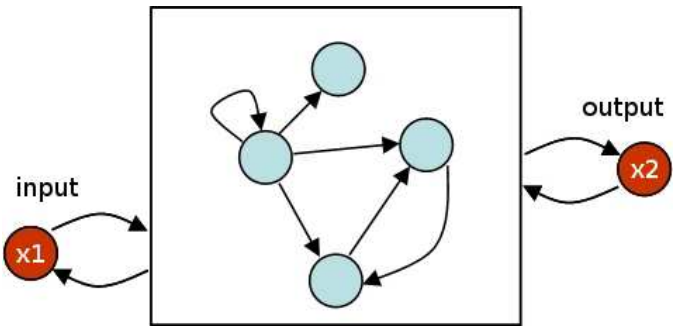

Fig. 1. An arbitrary GRN with two genes: x_1 and x_2 . Here, x_1 and x_2 are the input and the output node of the network respectively.

Consider an arbitrary simulated GRN inside a cell, with two genes (Figure 1). We denote the state (i.e., the activation level) of these two genes by x_1 and x_2, respectively; together the activation levels form the vector $\mathbf{X} = (x_1, x_2)$. The temporal behavior of any deterministic simulation of a regulatory network containing these two genes can now be described with respect to $\mathbf{X}$ by the differential equation

$$\mathrm{d}\mathbf{X}/\mathrm{d}t = \mathbf{F}(\mathbf{X}, \lambda, t), \tag{1}$$

where $\mathbf{F}$ is a vector field and λ is a vector of parameters. In this paper, we will focus on isolated cells in constant environmental conditions, such that $\mathbf{F} = \mathbf{F}(\mathbf{X}, \lambda)$. Hence, $\mathbf{F}$ describes a time independent, two dimensional vector for each system state $\mathbf{X}$, which represents the direction and magnitude of change in time, whenever the system reaches the state $\mathbf{X}$. $\mathbf{F}(\mathbf{X}, \lambda)$ is a vector field, which is referred to as the phase space of the dynamical system [16]. Vector field editing relies on creating and changing vector fields by superposition and adaptation of basic field elements $\mathbf{E}_i(X, \lambda_i)$. The vector field for any system state $\mathbf{X}$ is then given by the superposition of basic field elements:

$$\mathbf{F}(\mathbf{X}, \boldsymbol{\lambda}) = \sum_i \mathbf{E}_i(\mathbf{X}, \boldsymbol{\lambda}_i). \tag{2}$$

Typical elements have been proposed in [13] and [14] and can be grouped into singular elements and regular elements. Singular elements create a singularity in the vector field (i.e., a source or a sink) while regular elements in general do not. Two examples are depicted in Figure 2.

In our framework, we adopt the regular element formulation given in [13] and use a simplified version of singular elements. The regular element we use is called attachment element. It creates a flow of surrounding system states toward an attachment line at the center of the phase space. The mathematical formulation to create such an element, where the attachment line is oriented along an arbitrary angle $\theta \in [0, 2\pi]$ is given by

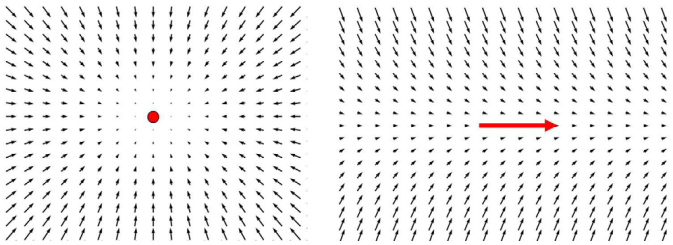

Fig. 2. Two basic field elements, which are also employed in the simulations: a singular element is depicted on the left panel, and a regular element (attachment element) is depicted on the right panel. Point and arrow mark the center and center line of the elements respectively.

$$\mathbf{A}(x_1, x_2) = \left(\begin{pmatrix} \cos\theta \\ \sin\theta \end{pmatrix} - cP(x_1, x_2) \begin{pmatrix} -\sin\theta \\ \cos\theta \end{pmatrix} \right). \quad (3)$$

Here, $P(x_1, x_2) = \sin\theta(x_1\, u_1) + \cos\theta(x_2\, u_2)$ and c is a parameter describing the speed with which the flow is attracted to the line and $\mathbf{U} = (u_1, u_2)$ is the center position of the element. Note, that for negative c, system states will diverge from the line instead of converging to it. To spatially limit the element's influence for superposition, this attachment element is multiplied by a Gaussian kernel $B(x_1, x_2)$ of width 2σ and center $\mathbf{U}$: $B(x_1, x_2) = e((x_1 u_1) + (x_2 u_2))/2\sigma$. Therefore, the complete formulation of the attachment element is given by

$$\mathbf{V}^R = B(x_1, x_2) \cdot \mathbf{A}(x_1, x_2). \quad (4)$$

We create a singular element by applying

$$\mathbf{V}^S(x_1, x_2) = \begin{cases} (\mathbf{U} - \mathbf{X})/\sigma & \text{if } r < \sigma \\ (2/r - 1/\sigma) \cdot (\mathbf{U} - \mathbf{X}) & \text{if } \sigma \leq r < 2\sigma. \end{cases} \quad (5)$$

The variable $r := ||\mathbf{X}\,\mathbf{U}||_2$ describes the distance of the system state $\mathbf{X}$ to the center $\mathbf{U}$ of the singular element. The width of the element is denoted by 2σ. A superposition of η field elements, each weighted by a factor α_i, yields an arbitrarily complex vector field, which can be interpreted as the system phase space:

$$\mathbf{F}(x_1, x_2) = \sum_{i=1}^{\eta} \alpha_i \mathbf{V}_i(x_1, x_2), \quad (6)$$

where $\alpha_i \mathbf{V}_i(x_1, x_2)$ corresponds to $\mathbf{E}_i(\mathbf{X}, \lambda_i)$ in Equation (2), with λ consisting of all $\mathbf{U}_i$,σ_i ,α_i of all field elements, and additionally θ_i and c_i of the regular elements. Thus, the vector field described in Equation (6) constitutes the right hand side of the differential equation

$$\frac{\mathrm{d}\mathbf{X}}{\mathrm{d}t} = \mathbf{F}(\mathbf{X}, \boldsymbol{\lambda}), \quad (7)$$

which is integrated from $t = 0$ to $t = t_{max}$ to yield a trajectory of the dynamical system. Evolving such a system consists of representing λ in a chromosome and employing standard evolutionary methods for the search, such as evolutionary strategies (ES) [17].

III. A Prerequisite: Strong Causality and the Genotype to Phenotype Map

Evolvability in evolutionary strategies has been investigated in [11]. The authors proposed that a mutation operator should preserve the neighborhood structure in corresponding evolutionary spaces, i.e., the genotype space, the phenotype space, (and possibly the fitness space). They state that this is a prerequisite for a successful evolutionary search, and quantify the so-called causality condition in terms of probabilities: $\forall g_i, g_j, g_k$:

$$\begin{gathered} \|f(g_i) - f(g_j)\| < \|f(g_i) - f(g_k)\| \\ \Leftrightarrow \\ P(g_i \to g_j) > P(g_i \to g_k) \end{gathered} \quad (8)$$

Here, g_i, g_j, g_k denote arbitrary genotypes while $f(*)$ is the mapping between genotype and phenotype spaces. $P(g_a \to g_b)$ is the probability for a mutation to result in a transition from genotype g_a to genotype g_b. In [11], it has been shown that for a direct genotype-phenotype mapping, evolutionary strategies inherently possess strong causality since the mutation distribution is Gaussian with zero mean. Thus, in an evolutionary strategy with an indirect mapping to maintain this causality, it would obviously be beneficial to have a high covariance between changes in genotypes and changes in phenotypes. For maximal covariance, a monotonic relation between genotypic and phenotypic changes will allow an evolutionary strategy to satisfy condition (8).

We will investigate the estimated change in causality of the genotype to phenotype mapping created by both the graph based developmental control represented by the CTRNN, as well as the VFE based developmental control. To do so, we will firstly define a phenotype representation for dynamical systems, which forms a metric space, and thus allows the calculation of $\|f(g_i) - f(g_j)\|$ in condition (8). Subsequently, we will compare the covariance of VFE and graph representations to gain insight into the evolvability of both approaches.

IV. A Phenotype for Dynamical Systems

The way we represent the phenotype of a dynamical system is conceptually similar to [18]. It is not straight forward to define a general "phenotype" on the meta-level of dynamics. However, as we have seen in the context of development, the dynamics of a system are used to control cellular growth. Depending on the interpretation of dynamical features, system transients or trajectory endpoints (i.e. stable attractors) seem suitable for a representation of the phenotype. Since in the majority of biological organisms, thresholds in the genetic regulation of cellular processes combined with transient behavior are used to control development, we believe that the phenotype representation should take this into account and should be able to capture transient-related features and changes thereof. In the following, we will introduce discrete fields, as possible phenotype for dynamical systems.

A. Discrete Fields

Evaluating the vector field in the general dynamical systems Equation (1) at n regular grid points in all m cardinal directions yields n^m vectors $\mathbf{v}_l, l \in \{1, ..., n^m\}$ that approximate the dynamical system in a discretized way. A change in any of the vectors $\mathbf{v}_l$ can be seen as a change in parts of the transients of the system, since it represents a local change in phase space, which does not necessarily result in a change of stable attractors. Note, that this discrete representation can easily be computed for both CTRNNs and VFE. In Vector Field Embryogeny, the right hand side of Equation (2) gives $\mathbf{v}_l$ directly when evaluated at the n^m points. For graphs, initializing the nodes with the same n^m discretization points, and subsequent iteration of graph dynamics for one step yields the necessary data. The difference between each of the n^m system states after iteration and the corresponding initialization values yields a local derivative, which corresponds to the vectors $\mathbf{v}_l$.

B. Field Difference

The field difference $FD(A, B)$ between two dynamical systems A and B, represented by discrete fields $\mathbf{v}_l^A$ and $\mathbf{v}_l^B$ is given by the normalized sum of the Euclidean distances between corresponding vectors:

$$FD(A, B) = \frac{1}{n^m} \sum_{l=1}^{n^m} \left\| \mathbf{v}_l^A - \mathbf{v}_l^B \right\|_2 \tag{9}$$

In the following, we will use the field difference to determine $\|f(g_i) - f(g_j)\|$ in Equation (8), i.e., the phenotype difference of the two individuals i and j.

V. Experiments

A. Causality and Covariance in Graph Based Modeling

For the calculation of the field difference, three nodes of the CTRNN are selected randomly. Thus, the observed phase space is 3-dimensional, independent from the actual network size. Note, that this choice is arbitrary and will not change the general process. Reducing the observation of a network to a smaller number of nodes represents an investigation in the light of a developmental model, since usually not all nodes in a graph control development directly, but only a small subset is responsible for the interface to the cellular model. Each of the three dimensions is sampled with $n = 20$ discretization steps as points for the calculation of $\mathbf{v}_l$, thus, 8000 vectors represent each phase space. The initial activation value of all other neurons is drawn from a uniform random distribution in the range $[0, 1]$.

Firstly, we will visualize the characteristics of the different genotype to phenotype maps. To this end, we want to introduce $\Delta G \Delta P$-plots, which display the phenotypic change ΔP resulting from a number of genotypic changes ΔG. The left panel of Figure 3 shows a sample representation of a $\Delta G \Delta P$-plot, which is created in the following way[1]:

[1]All of the experiments in this paper are based on the statistical analysis of samples. The sample size is chosen as a trade-off between computational complexity and statistical variation, which is indicated in most figures.

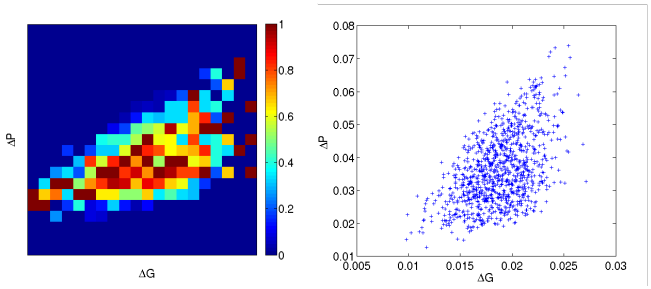

Fig. 3. An example for the $\Delta G \Delta P$ plots. The right panel shows the data sample that is used to create the plot in the left panel. The sample is taken from an experiment using simple neural networks with three nodes. Only one of the 100 $\Delta G \Delta P$ plots which are created for each experiment is shown.

1) Sample the genotype space with 100 random points $g_i^0, i \in \{1, ..., 100\}$.
2) Create the discrete fields $\mathbf{v}_l$ for these 100 points, and denote them as phenotypes $p_i^0, i \in \{1, ..., 100\}$.
3) Mutate each g_i^0 1000 times by adding uniformly distributed random variables in the range [-0.1,0.1] to receive genotypes $g_i^j, j \in \{1, ..., 1000\}$. This way the genotype neighborhood in the 100 genotype locations is sampled.
4) Then, calculate new phenotypes $p_i^j, j \in \{1, ..., 1000\}$.
5) For each of these, calculate the genotype distance $\Delta G_i^j = \frac{1}{N} \left\| g_i^0 - g_i^j \right\|_2$, where N is the number of entries in the genotype vectors. Note, that for evolutionary strategies, this genotype distance correlates with the probability of transition from g_i^0 to g_i^j.
6) Calculate the phenotype distance as field difference $\Delta P_i^j = FD(p_i^0, p_i^j)$.
7) The resulting pairs $\Delta P_i^j \Delta G_i^j$ can be visualized in a histogram fashion to characterize the mapping. These plots are constructed as a 2D histogram, where the widths of the 20×20 bins are chosen to span over the range $[\min_{i,j}(\Delta G_i^j), \max_{i,j}(\Delta G_i^j)]$ along the ΔG-axis, and over the range $[0, \max_{i,j}(\Delta P_i^j)]$ along the ΔP-axis. In such plots, bin columns contain all ΔP-values that result from a fixed ΔG-range. Each bin column is normalized individually, such that, for each column, the highest bin has value 1. This accounts for a possibly uneven distribution of mutations along the ΔG axis.

Such plots give first information on the mutational behavior of the genotype to phenotype map. Also, because of the sampling of the genotype space, we can infer homogeneity of the mutational behavior of the genotype to phenotype map. As an example, Figures 3 and 4 show representative plots for the mutations of one point g_i^0, for graphs of size 3 and 20, respectively. The 20 nodes setup exhibits cloud-like distributions of $\Delta G \Delta P$ pairs. Independent of the mutation strength, both small and large changes in the genotype can be achieved. Note, that this behavior can be observed for all sampled g_i^0 (data not shown).

From the plots alone, we can observe that the covariance between ΔG and ΔP is higher for graphs with a low number

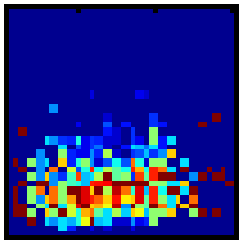

Fig. 4. A second example for the $\Delta G \Delta P$ plots. The figure shows a representative plot for neural networks with 20 nodes.

of nodes than for larger graphs. Remember that for a mapping to maintain causality in ES, a distribution in the $\Delta G \Delta P$ plots that rises monotonically would be ideal; this would correspond to a high covariance in the data points. Therefore, we choose the covariance between ΔG and ΔP for different representations to estimate the strong causality. For a systematic comparison, we subtract the mean and normalize the data for each of the 100 sets of data points that define the $\Delta G \Delta P$ plots to $[-1, 1]$. Note, that normalization is sensible, since ES self-adapts mutational step sizes, such that the qualitative, and not the quantitative relation between ΔG and ΔP values are important.

Figure 5 gives the distributions of the covariances for the 100 g_i^0, evaluated in differently sized networks. The plot verifies the observations above: the covariance rises with smaller network sizes. Thus, for evolutionary strategies, we could expect that only very simple graph structures can be evolved efficiently for developmental control.

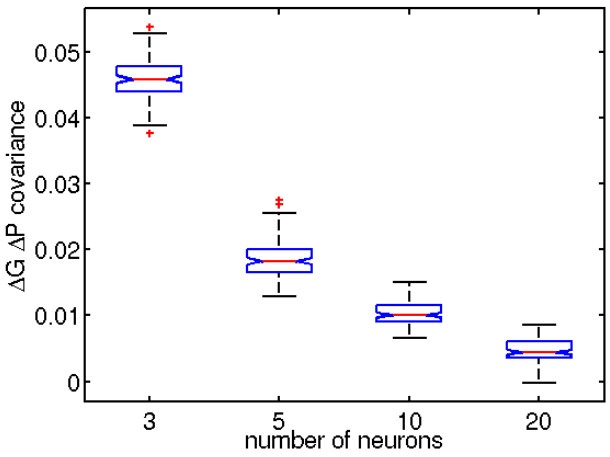

Fig. 5. The plot shows the distributions of covariances for graph setups with different numbers of nodes. Each box represents the data of 100 genotype positions g_i^0.

B. Causality and Covariance in Vector Field Embryogeny Based Modeling

The same investigations can now be performed for the Vector Field Embryogeny representation. Similar to the graph experiment, Figure 6 shows a representative $\Delta G \Delta P$ plot for VFE, which employs 6 basic field elements. We can observe a

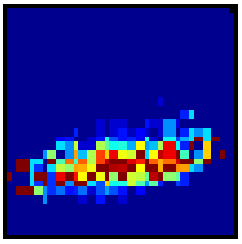

Fig. 6. A representative $\Delta G \Delta P$ plot for a Vector Field Embryogeny representation with 6 basic field elements is shown.

slightly positive slope of the distribution. Again, Figure 7 gives the distributions of the covariances for the 100 g_i^0, evaluated for different numbers of basic field elements. We can see

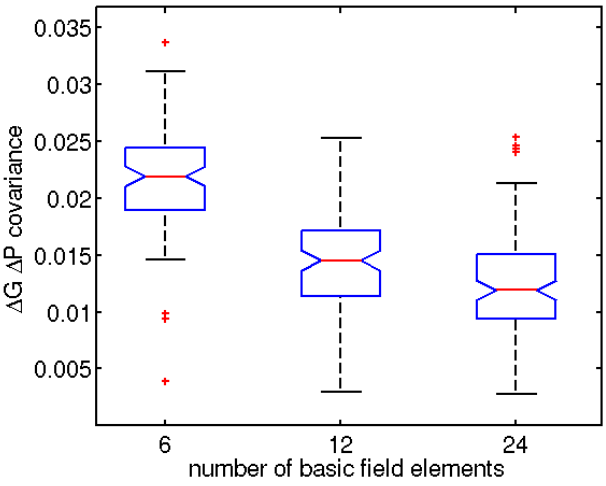

Fig. 7. The plot shows the distributions of covariances for VFE with different numbers of nodes. Each box represents the data of 100 genotype positions g_i^0.

a similar trend as for the graph representations: covariances decrease for larger systems.

VI. Comparison of CTRNN and VFE Representations

To be able to compare the two approaches for developmental control, causality and covariance only represent the capability for evolutionary computation to advance via stabilized mutations. However, it is also necessary to compare systems that have the same representational power, i.e., the same capabilities to represent different dynamics.

Typical features of dynamical systems such as the number of attractors and the basins of attraction are interesting for the analysis of stable system states, but neglect most of the transient features. At the same time, evaluating transients yields highly variable, system dependent information. An illustrative example is the comparison of two similar phase spaces, except

that the one is scaled by a factor of two, i.e. the system is twice as fast. For the attractor based investigations, both would obviously yield the same results, while for a transient based investigation, they would be different. Therefore, we need to decide whether for our purpose the transient or the attractor based view is most appropriate.

We are looking for a measure of representational power that is adapted to the use of dynamical systems for controlling development. In developmental models the transient often works as a switch, e.g. cell division is triggered if the concentration of a certain chemical (transcription factor) exceeds a threshold. Therefore, the transient behavior of the control system is important for our investigations. Furthermore, as mentioned above, not all state variables of a dynamical system directly influence the cellular behavior in developmental modeling. Therefore, we will assume that the transients of only a restricted number of state variables of the dynamical system is of interest for determining the representational power. The aforementioned thresholds are usually specified in the genome during the evolutionary process. Since they are individually defined for single chemicals they can be regarded as divisions of the phase-space, perpendicularly to its cardinal directions. If a trajectory of a system state crosses such a threshold, an action is triggered. When a threshold exists for each cardinal direction, it is possible to constrain a cellular function to an arbitrary rectangular subvolume of the phase space. Therefore, the ability of a dynamical system to reach these subvolumes during the transient behavior is closely linked to its ability to represent rich differential cellular activity patterns. Thus, we measure the representational power of a dynamical control system for development by the fraction of the phase space it can reach during the transient behavior. Note, that this measure intentionally ignores the attractor structure of the dynamical system. In the following, we will investigate the covered phase space, i.e., the area in the phase space that can be reached by the dynamical system from predefined starting points.

The experiments are structured as follows: We initialize nine system starting states $\mathbf{X}^i = (x^i, y^i, z^i), i \in \{1, ..., 9\}$ in a three dimensional phase space at $x^i \in \{0, 0.5, 1\}$ and $y^i \in \{0, 0.5, 1\}$, where all $z^i = 0.5$. From a developmental point of view, this could represent the system state of nine undifferentiated cells with distinct maternal gradient information. Here the initial states of genes x and y are determined by the system configuration, e.g. maternal gradients and the state of gene z would represent a cellular differentiation variable. The dynamical system is initialized with a random chromosome, and from each of these starting points, state space trajectories are iterated for 100 time steps. All $9 \cdot 100$ phase space positions visited by the system states are recorded. Then, the procedure is repeated 999 times with a new random chromosome. In this way, the resulting point-cloud of all transient system states of the 1000 random samples from the genotype space gives an estimate of the phase-space coverage of the representation. In order to calculate the fraction of the covered phase space, we have to discretize the space and check for each sub-volume whether it has been visited by a transient or not. Therefore, we subdivide the phase space into $10 \times 10 \times 10$ equally sized sub-volumes, and check for each if they contain at least one of the previously recorded system states. Then, we can count these sub-volumes and get an estimate of the representational power of the dynamical system.

For Vector Field Embryogeny, Figure 8(a) gives a plot of the fraction of the covered phase space over the number of basic field elements, i.e., the degrees of freedom employed. Note, that we chose half of the number of basic field elements to be singular, and the other half to be regular elements, similar to the other VFE-setups in this paper. We can see that phase space coverage is 100% for lower degrees of freedom. Interestingly, the phase space coverage decreases steadily with higher numbers of basic elements. While at first this seems counter-intuitive, we can expect that for a large number of basic field elements, it is likely that a strong attractor is close to the starting point of the system states. In this case, the trajectory will quickly converge to the attractor state and phase space coverage will be small. This is supported by the observation that the covered phase space in these cases is located exclusively in the vicinity of the starting points (data not shown).

Figure 8(b) gives a plot of the fraction of covered phase space over different graph sizes for the CTRNN representation. We can see that for a low number of nodes, less than half of the phase space is covered by the point cloud. With an increasing number of nodes, phase space coverage increases to over 87% since the representation represents richer dynamics. However, for larger graphs, the phase space coverage decreases, where the largest network of 96 nodes covers only 30%. This decreasing phase space coverage with increasing degree of freedom of the dynamical system has equally been observed for the VFE framework. We argued that this effect could be due to attractors lying close to the starting points. Tt is conceivable that a similar effect occurs for graph-based representations, however, this remains to be investigated in more detail.

The two plots in Figure 8 allow us to assess representational power of the two approaches and give us a basis for a fair comparison of evolvability. As an example, we choose a graph with 20 nodes (400 degrees of freedom), which gives a phase space coverage of 87% (represented by the circle in Figure 8(b)). From the circles shown in Figure 8(a), we can see that a Vector Field Embryogeny representation using 12 basic field elements (78 degrees of freedom) clearly has a higher phase-space coverage, while a VFE using 24 basic field elements (156 degrees of freedom) clearly has a lower phase space coverage than the chosen graph sample.

Figure 9 shows the $\Delta G \Delta P$ covariance values for these three cases: a CTRNN graph with 20 nodes, a Vector Field Embryogeny representation with 24 basic elements, and one with 12 basic elements. We can see that covariance values are low in general, but are significantly higher for both Vector Field Embryogeny representations.

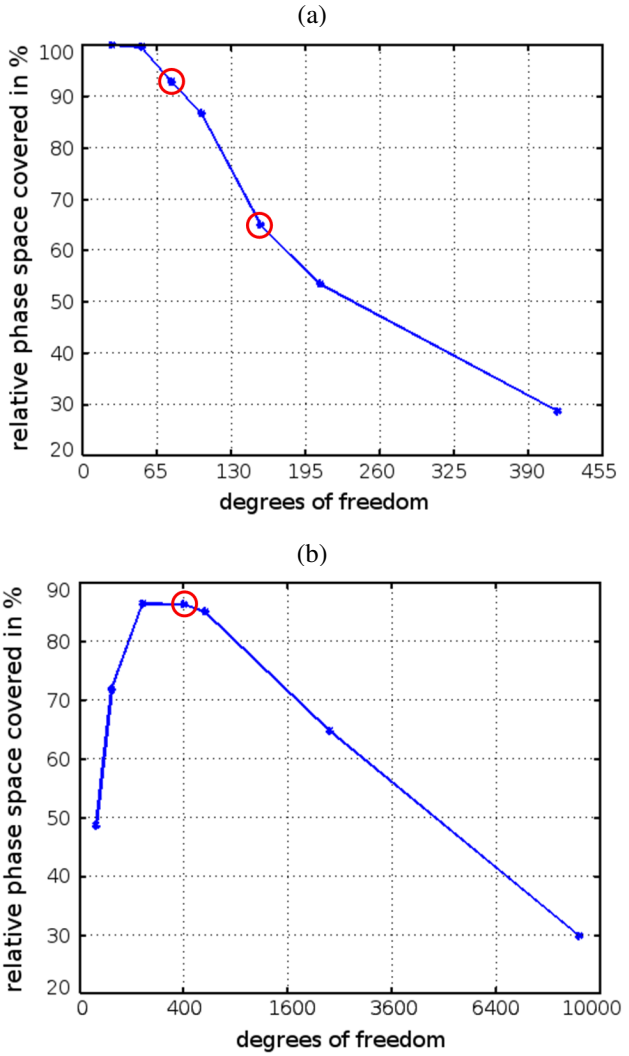

Fig. 8. The fraction of phase space coverage of VFE (panel (a)) and CTRNN (panel (b)) Note, the abscissa values, giving the dimensionality of the search space: in (b), for a fully connected graph, the number of nodes is squared to determine the number of free parameters. In (a), the number of basic field elements is multiplied by the number of parameters that encode each element. Highlighted by circles are the three setups that are used for comparison (see text for more details).

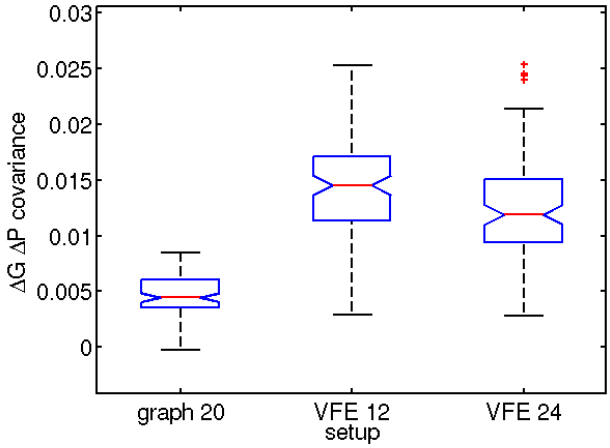

Fig. 9. Covariance values for CTRNN graph and vector field representations against numbers of nodes and basic field elements. Each box represents the data of 100 genotype positions g_i^0.

VII. Discussion

Developmental representations in evolutionary computation have the potential to enable the optimization of complex systems. At the same time, indirect representations make evolutionary algorithms more difficult to analyze, e.g. the mutational effect on the phenotype level, which is important for guiding the search process and the self-adaptation of parameters in particular for evolutionary strategies, is neither easy to determine nor to control.

The $\Delta G \Delta P$ plots introduced in this paper provide a visual cue on the mutational effect and the covariance between changes on the genotype and the phenotype space. In evolutionary strategies, the covariance of the $\Delta G \Delta P$ data can be used as an indicator of evolvability. The experiments have shown that for CTRNN graphs, covariance decreases with increasing number of nodes. For Vector Field Embryogeny, this trend can also be observed. This strongly motivates the use of an incremental approach toward more complex systems, where starting from a small graph or from a vector field representation with only a few basic field elements, evolution has the ability to add nodes or field elements when progress stagnates. NEAT [19] is such an approach for graphs; similar ideas could be envisioned for vector fields.

When comparing the evolvability of different representations, we have to take their respective representational power into account. As a first approximation we have chosen to estimate the number of cellular functions that could be represented by the fraction of discretized phase space volume which sensibly initialized trajectories could reach. This provides us not only with a basis for a fair comparison but also gives us additional insight into the dynamical control system that is represented by vector field and graph-based models.

At first sight it has been surprising that the proposed measure for the representational power decreased with increasing degree of freedom for both the graph and the vector field based models. In VFE, basic flow field features such point attractors are directly manipulated and it seems likely that by increasing the number of point attractors in a phase space of restricted size, trajectories will become shorter and the covered phase space volume smaller. For the continuous time recurrent neural networks, an increase of the number of nodes cannot be directly associated with an increase of point attractors and therefore, the results are more difficult to interpret. Furthermore, Figure 8(b) shows that the representational power does not monotonously decrease but reaches a maximum at around 250 degrees of freedom. Although it is known that larger graphs potentially have a higher number of attractors [20], it remains to be investigated whether this can explain our results in general.

For VFE, this finding would again motivate an incremental approach, with careful introduction of new basic field elements. For example, a new random element could be added with zero weight initialization such that it does not change the phase space at first. Evolution could then tune the weight and thereby increase the complexity of the representation incrementally. Such adaptive representations with neutral mutations to increase the degree of freedom of the representation without changing the search space have been successfully employed in design optimization [21].

When we choose a graph and a vector field representation with similar phase space coverage, we find that Vector Field Embryogeny has a higher covariance between phenotypic changes and genotypic changes leading to a higher degree of causality than CTRNN. For evolutionary strategies, this correlates with a higher success rate of Vector Field Embryogeny in solving cellular differentiation tasks. The observed covariance values are low in general. We expect to find other features that distinguish the two dynamics representations and additionally account for the observed higher evolvability of VFE. In future, we will try to alter VFE and graph representations to increase covariances in both without reducing representational power. This will allow us to assess covariance of $\Delta G \Delta P$ as useful evolvability measure in evolutionary strategy frameworks.

Another interesting fact from an evolutionary perspective is the dimensionality of the genotype space for such matched representations: while the genotype space grows quadratically with the number of nodes in a fully connected graph, it grows only linearly with the number of basic field elements in Vector Field Embryogeny. This could naturally alleviate the search process in more complex domains.

Our analysis can be extended to dynamical systems that represent coupled cellular states, such as diffusion based multicellular systems. Performing the described process in parallel for each cell of an individual would yield n phase spaces and field differences for n cells which would simply increase the sample for statistics.

The investigations above have been performed using simple basic field elements for VFE only. We think that one of the strengths of VFE lies in the possibility to modularly choose more advanced, non-trivial basic field elements for a problem-adapted search space. Apart from *a posteriori* analysis, the causality considerations in this paper could also be used as an *a priori* benchmark to assess evolvability of more complex basic field elements.

REFERENCES

[1] K. Stanley and R. Miikkulainen, "A taxonomy for artificial embryogeny," *Artificial Life*, vol. 9, pp. 93–130, 2003.

[2] P. Hogeweg, "Evolving mechanisms of morphogenesis: on the interplay between differential adhesion and cell differentiation," *Journal of Theoretical Biology*, vol. 203, pp. 317–333, 2000.

[3] T. Kowaliw, "A good number of forms fairly beautyful," Ph.D. dissertation, Department of Computer Science, Concordia University, Montreal, Canada, 2007.

[4] J. 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. Bentley, Eds. Elsevier, 2003, pp. 278 –301.

[5] J. Rieffel and J. Pollack, "Automated assembly as situated development: Using artificial ontogenies to evolve buildable 3-d objects," in *Proceedings of the GECCO'05*, 2005.

[6] D. Roggen, D. Federici, and D. Floreano, "Evolutionary morphogenesis for multi-cellular systems," *Genetic Programming and Evolvable Machines*, vol. 8, pp. 61–96, 2007.

[7] T. Steiner, J. Trommler, M. Brenn, Y. Jin, and B. Sendhoff, "Global shape with morphogen gradients and motile polarized cells." in *Proceedings of the 2009 Congress on Evolutionary Computation*, 2009, pp. 2225–2232.

[8] G. Tufte, "Gene regulation mechanisms introduced in the evaluation criteria for a hardware cellular development system." in *Proceedings of the First NASA/ESA Conference on Adaptive Hardware and Systems*, 2006, pp. 137–144.

[9] T. Steiner, Y. Jin, and B. Sendhoff, "Evolving heterochrony for cellular differentiation using vector field embryogeny," in *Proceedings of the 12th Annual Conference on Genetic and Evolutionary Computation*, 2010, pp. 571–578.

[10] ——, "Vector field embryogeny," *PLoS ONE*, vol. 4, no. 12, pp. 1–16, 2009.

[11] B. Sendhoff, M. Kreutz, and W. von Seelen, "A condition for the genotype-phenotype mapping: Causality," in *Proceedings of the Seventh International Conference on Genetic Algorithms (ICGA '97)*, 1997, pp. 73–80.

[12] T. Jones and S. Forrest, "Fitness distance correlation as a measure of problem difficulty for genetic algorithms," in *Genetic Algorithms: Proceedings of the 6th International Conferences (ICGA)*, L. Eshelman, Ed. Morgan Kaufmann, 1995, pp. 184–192.

[13] G. Chen, K. Mischaikow, R. Laramee, P. Pilarczyk, and E. Zhang, "Vector field editing and periodic orbit extraction using morse decomposition." *IEEE Transactions on Visualization and Computer Graphics*, vol. 13, no. 4, pp. 1077–2626, 2007.

[14] E. Zhang, K. Mischaikow, and G. Turk, "Vector field design on surfaces." *ACM Transactions on Graphics*, vol. 25, no. 4, pp. 1294–1326, 2006.

[15] E. Praun, H. Hoppe, M. Webb, and A. Finkelstein, "Real-time hatching." in *Proceedings of the ACM SIGGRAPH*. ACM, 2001, pp. 581–586.

[16] G. Nicolis, *Introduction to Nonlinear Science.* Cambridge University Press, 1995.

[17] H.-P. Schwefel, *Evolution and Optimum Seeking*. Wiley-Interscience, 1995.

[18] Y. Jin and J. Trommler, "Ga fitness-independent evolvability measure for evolutionary developmental systems," in *Proceedings of the IEEE Symposium on Computational Intelligence in Bioinformatics and Computational Biology*, 2010.

[19] K. Stanley, "Efficient evolution of neural networks through complexification," Ph.D. dissertation, Artificial Intelligence Laboratory, The University of Texas at Austin, 2004.

[20] A. Knoblauch, G. Palm, and F. Sommer, "Memory capacities for synaptic and structural plasticity," *Neural Computation*, vol. 22, pp. 289–341, 2010.

[21] M. Olhofer, Y. Jin, and B. Sendhoff, "Adaptive encoding for aerodynamic shape optimization using evolution strategies," in *Congress on Evolutionary Computation (CEC)*, vol. 2, Seoul, Korea, May 2001, pp. 576–583.