Theme Preservation and the Evolution of Representation

Keki Burjorjee and Jordan Pollack

Demo Lab,
Computer Science Department MS018,
Brandeis University,
Waltham, MA 02454, USA
{kekib, pollack}@cs.brandeis.edu

Abstract. The identification of mechanisms by which constraints on phenotypic variability are tuned in nature, and the implementation of these mechanisms in Evolutionary Algorithms (EAs) carries the promise of making EAs less "wasteful". The constraints on phenotypic variability are determined by the way genotypic variability maps to phenotypic variability. This in turn is determined by the way that phenotypes are *represented* genotypically. We use a formal model of an EA to show that when some part of the genome is mutated with a much lower probability than some other part, representations used to search the phenotype space - and hence the constraints on phenotypic variability - can themselves be thought to evolve. Specifically, we formally analyze a class of mutation-only fitness proportional evolutionary algorithms and show that these evolutionary algorithms implicitly implement what we call *subrepresentation evolving multithreaded evolution*. These EAs conduct second-order search over a predetermined set of representations and exploit promising representations within this set for first order evolutionary search. We compare our analytical method and results with those employed in schema analysis and note that by examining systems that are simpler than the ones examined in a typical schema analysis (mutation is the only variational operator in our systems), and by changing how we define the subsets of the genotype space that are analyzed, we have obtained results that are more intuitively understandable and are *not* specific to a particular data-structure. [1]

1 Introduction

Phenotypic variability, whether in natural or artificial evolution, is always constrained - a sunflower cannot be the offspring of an elephant, a travelling salesman path does not mutate with equal probability into all other TSP paths. An

[1] This paper is similar to an identically titled workshop paper [2] (see http://demo.cs.brandeis.edu/papers/burjorjee05.pdf). The use of the Kullback-Liebler divergence operator in this paper simplifies the proof of our main result. Additionally, in this paper we illuminate our analytical method and results by comparing them with those in other work on schema theory.

 ©IICAI

important observation is that many of the constraints on phenotypic variability in natural evolution are *not* arbitrary. They are, instead, sensitive to the problem domain. To understand what we mean by this consider the variation present in a litter of offspring of two healthy mammals. One constraint on this variation is external symmetry - the bodies of all offspring are constrained to be roughly symmetrical. The opposite is true in the plant kingdom - typically none of the offspring of plants are symmetrical. Mammals require symmetry for efficient locomotion (almost every locomoting machine that man has created has a roughly symmetrical body), on the other hand plants do not locomote, so symmetry is not important for their survival. Symmetry is thus an example of a constraint on the phenotypic variability of mammals that is sensitive to the problem domain. For brevity's sake, when the constraints on phenotypic variability are sensitive to the problem domain we shall say that the phenotypic variability is *tuned to* the problem domain.

Let us think of higher fitness as an increase in the satisfaction of domain-sensitive phenotypic constraints (such as symmetry). Then if phenotypic variability becomes tuned to a problem domain, evolutionary search becomes more and more focused on phenotypes with higher fitness. Search is thus much less "wasteful" than if the constraints on phenotypic variation are arbitrary. For example, the evolution of mammals would probably have been much more wasteful, i.e. inefficient, if only one in a thousand offspring is symmetric.

Many researchers in the field of evolutionary computation (EC) see natural evolution as a search process and are interested in extracting its core algorithmic essence in order to construct efficient search algorithms that can be applied to difficult real world search problems. To this end those aspects of natural evolution which are thought to be essential to effective search are identified, abstracted and implemented in Evolutionary Algorithms (EAs). (e.g. genotype space, G-P map, fitness function, variation and selection) The phenotypes of most EAs do not however end up having variability that is tuned to the problem domain - in later generations the majority of the offspring of fit individuals are typically much worse than their parents. Biological offspring, on the other hand, tend to have approximately the same fitness as their parents.

The tuning of phenotypic variability in nature is not a "frozen accident from life's origins" [10] but is itself the product of certain mechanisms inherent within natural evolution. The identification of these mechanisms, and their implementation in EAs carries the promise of making EAs less "wasteful" and hence more efficient. In this paper we identify one such mechanism.

The constraints on phenotypic variability are determined by the way genotypic variability maps to phenotypic variability [10]. This in turn is determined by the way that phenotypes are *represented* genotypically - different genetic representations of some phenotype induce different constraints on the variability of that phenotype [8].

We use a formal model of an EA to show that when some part of the genome is mutated with a much lower probability than some other part, representations used to search the phenotype space - and hence the constraints

 ©IICAI

on phenotypic variability - can themselves be thought to evolve. Specifically, we formally analyze a class of mutation-only fitness proportional evolutionary algorithms and show that these evolutionary algorithms implicitly implement what we call *subrepresentation evolving multithreaded evolution*, i.e. these EAs conduct second-order search over a predetermined set of representations and exploit promising representations within this set for first order evolutionary search.

Our theory is developed in sections 2-6. We discuss the results in section 7, and in section 8 we compare our analytical method and results with the methods and results found in a typical schema analysis e.g. [4, 5, 7]. We conclude that section by noting that by examining systems that are simpler than the ones examined in a typical schema analysis and by changing how we define the subsets of the genotype space that are analyzed, we have obtained results that are more intuitively understandable and are *not* specific to a particular data-structure.

2 Mathematical Preliminaries

All sequences in this paper are zero based (the index of the first element is zero) and infinite. Let X be some set. Then we denote some sequence of elements in X by $\{p^n\}_{n\geq 0}$. For any $i \in \mathbb{N}$, we denote the element with index i in $\{p^n\}_{n\geq 0}$ by p^i. For some sets X, Y, and some function $\gamma : X \to Y$, we use the notation $\langle y \rangle_\gamma$ to denote the set $\{x \in X \,|\, \gamma(x) = y\}$. We will drop the subscript γ from this notation when it is clear from the context.

As in [8], for any set X we use the notation Λ^X to denote the set of all probability distributions over X, i.e. Λ^X denotes set $\{f : X \to [0,1] \mid \sum_{x\in X} f(x) = 1\}$.

We extend this notation to denote the set of all 1-parent (i.e. mutation-only) transmission functions (see [1]) over some set as follows: for any set X, the set of all 1-parent transmission functions $\{f : X \times X \to [0,1] \mid \forall x' \in X, \sum_{x\in X} f(x,x') = 1\}$ is denoted by Λ_1^X. Employing the notation used in [8], we use conditional probability notation to denote a 1-parent transmission function (henceforth transmission function). Thus a transmission function $f(x,x')$ is denoted $f(x|x')$.

3 Themes

Let G be some set of genotypes and let K be some set of objects that codify "properties" that are possessed by elements of G. If the properties are such that every genotype in G possess one and only one property in K (the properties are mutually exclusive), then we say that K is a theme set of G. We call the properties in K themes, and for any $k \in K$ we call the subset of genotypes that map to k the theme class of k. While we could have expressed these ideas in the terminology of mathematical functions, we believe that the terminology just introduced is better suited to our exposition. The correspondence between function terminology and our terminology is made clear in the following definition.

 ©IICAI

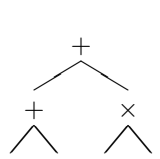
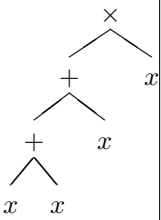
Genotype Set G	Theme set K	Sample Parent	Sample Child	Theme Preserved
Bitstrings in $\{0,1\}^{10}$	Strings in $\{0,1\}^4??????$	1011010111	1011100011	1011??????
S-expressions with binary tree structures that express single variable polynomials, i.e. binary trees with leafs x and internal nodes drawn from $\{+,\times\}$	Any $k \in K$ is a multiset of elements from $\{x,+,\times\}$ s.t. there exists a binary tree structure which uses all the elements in k			The multiset $\{\times,+,+, x,x,x,x\}$
Seeded L-Systems with terminals drawn from the alphabet $\Sigma = \{a,b,c,d\}$	The set of all seeded L-System "skeletons"	$[\langle abbcbd\rangle$ $(a \to b)$ $(b \to bcab)$ $(c \to ac)]$	$[\langle acbdbd\rangle$ $(a \to a)$ $(b \to ccab)$ $(c \to bc)]$	$[\langle ??????\rangle$ $(a \to ?)$ $(b \to ????)$ $(c \to ??)]$

Table 1. Three different theme-preserving mutation operators

Definition 1. (Theme Map, Theme, Theme Set, Theme Class) *Let X, Y be sets and let $\beta : X \to Y$ be a function. We call β a theme map, call the co-domain Y of β a β-theme set, call an element of Y a β-theme, and call the preimage $\langle y\rangle$ of some $y \in Y$, the* β-theme class of y.

Remark 1. Given the objects defined above, it is easily seen that the set of all β-theme classes form a partition of X

The idea of a theme class is mathematically identical to the idea of a forma discussed in [6] - each of these objects is simply an equivalence class which belongs to the partition that is induced by some function. However the *application* of this mathematical idea in this paper differs in spirit from its application in [6]. There a forma describes some equivalence class of phenotypes, whereas in this paper a theme class is an equivalence class of genotypes.

3.1 Theme Preservation and Alteration

Let G, K be sets such that G is countable and let $\beta : G \to K$ be some function. For some mutation operator that operates on elements of G, we say that this operator is β-preserving if it leaves the β-themes of of its argument unchanged, i.e. the child produced by the mutation operator will always have the same β-theme as its parent. We say that the mutation operator is β-altering if it *always* changes the β-theme of its argument. Examples of theme-preserving mutation

 ©IICAI

operators for three different kinds of genotype data-structures — bitstrings, S-expressions and L-Systems are given below. The genotype set and theme set of each operator is described in the first two columns of table 1. The 3rd and 4th columns of the table schematically show the effect of the three mutation operators on three sample genotypes, and the last column schematically shows the theme that is preserved in each case. We leave it to the reader to think of the theme map, and a theme altering mutation operator in each case.

1. A mutation operator which operates on bitstrings of length ten and only modifies the last six bits of its argument.
2. A mutation operator which takes a S-expression for a polynomial as an argument and changes the tree structure of the S-expression in some way while leaving the values of the nodes unchanged.
3. A mutation operator which takes a seeded L-system[2] over the alphabet $\{a, b, c, d\}$ as its argument and *substitutes* a symbol for another symbol in the seed string and the right hand sides of the rewrite rules - it does not add or delete rewrite rules, does not change the number of symbols in a seed string or rewrite rule, and does not change the left hand side of the rewrite rules.

Theme preserving and altering mutation can be modelled by transmission functions with appropriate constraints.

Definition 2. (Preserving and Altering Transmission Functions) *Let X, Y be some sets, let $\beta : X \to Y$ be a function, let $M \in \Lambda_1^X$ be a transmission function. We say that M is* β-preserving *if*

$$\forall x, x' \in X, \quad \beta(x) \neq \beta(x') \Rightarrow M(x|x') = 0$$

and say that M is β-altering *if*

$$\forall x, x' \in X, \quad \beta(x) = \beta(x') \Rightarrow M(x|x') = 0$$

The following proposition gives us a useful property of a preserving transmission function.

Proposition 1. *Let X, Y be some sets, let $\beta : X \to Y$ be a function, let $M \in \Lambda_1^X$ be a β-theme-preserving transmission function. Then,*

$$\forall y \in Y, \forall x' \in \langle y \rangle \qquad \sum_{x \in \langle y \rangle} M(x|x') = 1$$

Proof. By definition of a transmission function we have that $\forall x' \in X, \sum_{x \in X} M(x|x') = 1$. But by (def 2), $\forall y \in Y$, $\forall x' \in \langle y \rangle$ and $\forall x \notin y$, $M(x|x') = 0$. Hence $\sum_{x \in \langle y \rangle} M(x|x') = 1$

[2] In this paper we call a tuple consisting of 1) an L-system over some alphabet Σ and 2) a string in Σ^*, a seeded L-System. The genotypes in the evolutionary algorithms in [8] are seeded L-Systems

 ©IICAI

4 Transmaps

In this paper we will focus our attention on EAs in which mutation is the only form of variation and selection is fitness proportional. We will call such EAs basic fitness proportional EAs or bfpEAs for short. The representation used by some bfpEA to search some set of objects is determined by 1) an alternate set of software objects called genotypes, 2) a function that maps genotypes to objects in the search space, and 3) the mutation operator that stochastically produces some child genotype given some parent genotype. A transmap, defined below, is our model for a representation.

Definition 3. (TRANSMAP) *A transmap is a 4-tuple* (G, P, ϕ, M) *such that* G *is a countable set called the genotype set,* P *is some set called the phenotype set,* $\phi : G \to P$ *is called the growth map, and* $M \in \Lambda_1^G$ *is a 1-parent transmission function.*

Let P be some set of objects (e.g. sorting networks, polynomial functions, plant morphologies, etc.) Then, given some representation for P we can construct a transmap B that models this representation as follows: 1) the search space P of the representation is the phenotype set of B, 2) the alternate set of software objects is the genotype set of B, 3) the function that maps the software objects to objects in the search space is the growth map of B, 4) the effect of the mutation operator of the representation is modelled by the transmission function of B.

4.1 Subtransmaps

In this section we show how, a transmap B with a β-preserving transmission function, determines a set of transmaps such that each transmap in this set is in one-to-one correspondence with some β-theme. The transmaps in this set are called subtransmaps of B.

For some function $f : X \to Y$, and some $A \subset X$, the restriction of f to A is denoted $f|_A$. We extend the notion of restriction to functions whose domain is the cross-product of the same set as follows: for a function $g : X \times X \to Y$, the restriction of g to A, denoted $g|_A$ is a function of type $A \times A \to Y$ such that for any $a_1, a_2 \in A$, $g|_A(a_1, a_2) = g(a_1, a_2)$.

Proposition 2. *Let* G, K *be sets, let* $\beta : G \to K$ *be some function and let* $M \in \Lambda_1^G$ *be a* β*-preserving transmission function. Then, for any* $k \in K$, $M|_{\langle k \rangle} \in \Lambda^{\langle k \rangle}$

Proof. For any k *in* K*, and any* $g' \in \langle k \rangle$, $\sum_{g \in G} M(g|g') = 1$. *But by* (def 2), $\forall g \notin \langle k \rangle$, $M(g|g') = 0$. *So,* $\sum_{g \in \langle k \rangle} M(g|g') = 1$, *which implies that* $\sum_{g \in \langle k \rangle} M|_{\langle k \rangle}(g|g') = 1$

Definition 4. ((β, k)-SUBTRANSMAP) *Let* $B = (G, P, \phi, M)$ *be a transmap s.t. for some set* K, $\beta : G \to K$ *is a function and* M *is* β*-preserving. We define the* (β, k)*-subtransmap of* B*, denoted* $B|_{\langle k \rangle}$*, to be the transmap* $(\langle k \rangle , P, \phi|_{\langle k \rangle} , M|_{\langle k \rangle})$.

 ©IICAI

To see that $(\langle k\rangle, P, \phi|_{\langle k\rangle}, M|_{\langle k\rangle})$ is indeed a transmap note that $M|_{\langle k\rangle} \in \Lambda_1^{\langle k\rangle}$ by proposition 2 and $\phi|_{\langle k\rangle}$ is of type $\langle k\rangle \to P$ by definition of restriction.

For some $\beta : G \to K$, suppose R is a representation with a mutation operator that preserves β, suppose B is a bfpEA which uses R and for some theme k, suppose all the genotypes in the initial population of B have the same theme $k \in K$. Then, as the mutation operator of B is β-preserving, all genotypes in all the generations of an evolutionary run of B will have theme k. Therefore, we can define a new representation R_k which is isomorphic to R over $\langle k\rangle$ by "pulling" the theme k out of the genotypes in $\langle k\rangle$ and "pushing" it into the mutation operator and growth function of R_k s.t. when B uses R and starts with a k-themed initial population, its search behavior is the same as if it used R_k and started with an isomorphic initial population in the genotype set of R_k. Let us call R_k a subrepresentation of R. Then a subtransmap, as defined above, is a model of a subrepresentation.

4.2 (β, ω)-Preserving Transmaps

Definition 5. (Rate Operator) *Let $M^1, M^2 \in \Lambda_1^G$ be transmission functions. For any $\omega \in [0,1]$ we define the* Rate Operator $\mathcal{R}^\omega : \Lambda_1^G \times \Lambda_1^G \to \Lambda_1^G$ *as follows:*

$$\forall g, g' \in G,\ \mathcal{R}^\omega(M^1, M^2)(g, g') = \omega M^1(g|g') + (1-\omega)M^2(g|g')$$

To see that for any set G and $\omega \in [0,1]$, the range of $\mathcal{R}^\omega$ is indeed Λ_1^G, observe that $\forall M^1, M^2 \in \Lambda_1^G, \forall g' \in G$,

$$\sum_{g\in G} \mathcal{R}^\omega(M^1, M^2)(g|g') = \omega \left(\sum_{g\in G} M^1(g|g')\right) + (1-\omega)\left(\sum_{g\in G} M^2(g|g')\right) = \omega + (1-\omega) = 1$$

Definition 6. ((β,ω)-preserving Transmap) *Let $B = (G, P, \phi, M)$ be a transmap, for some set K, let $\beta : G \to K$ be a function and let $\omega \in [0,1]$. B is said to be (β,ω)-preserving if there exists $M_1, M_2 \in \Lambda_1^G$ such that M^1 is β-preserving, M^2 is β-altering and $M = \mathcal{R}^\omega(M_1, M_2)$.*

Remark 2. Note that for any (β,ω)-preserving transmap (G, P, ϕ, M), the preserving and altering components of M (M_1 and M_2 in the definition above) are unique. We denote them as M_P and M_A respectively.

Suppose mutation in some representation is β-preserving with some probability ω and β-altering mutation with probability $(1-\omega)$, then it is easy to see how we could model such a representation using a (β,ω)-preserving transmap.

 ©IICAI

5 Evolution Machines

Definition 7. (Evolution Machine). *An evolution machine – which we also call an EM – is a 3-tuple (B, f, s) such that $B = (G, P, \phi, M)$ is a transmap, $f : P \to \mathbb{R}^+$ is called the fitness function and $s \in \Lambda^G$ is called the* initial genotype distribution.

An evolution machine is a collection of all the formal objects needed to model an evolutionary run of a bfpEA. A bfpEA, which was introduced in section 4, is similar to a Simple Genetic Algorithm as defined in [9] in all respects except that 1) it performs fitness proportional selection (an SGA may use other selection methods), 2) its genotypes may be instances of arbitrary datatypes (SGAs use only bitstrings), and 3) mutation is its only variational operator (SGAs also use a recombination operator).

Populations in a bfpEA are modelled as distributions of an EM. In order to define how these distributions change from generation to generation we recall, and extend, the following operators from [9] and [8].

Definition 8. (Selection Operator) *Let X be some set and let $f : X \to \mathbb{R}^+$ be some function. We define the* Selection Operator $\mathcal{S}_f : \Lambda^X \to \Lambda^X$ *as follows:*

$$(\mathcal{S}_f p)(x) = \frac{f(x)p(x)}{\sum\limits_{x' \in X} f(x')p(x')}$$

The selection operator is parameterized by a fitness function. It takes a distribution p_X over some set X as its argument and redistributes the probability mass of the distribution over the elements of X in proportion to the fitness of the elements and their probability mass in p_X. In typical usage of $\mathcal{S}$ in the literature, the set X is the genotype set. In this paper $\mathcal{S}$ will also be used to express meta-selection applied to a distribution over a theme set. The precise sense in which we use the phrase meta-selection will become clear later on.

Definition 9. (Expected Fitness Operator) *Let X be some set, and $f : X \to \mathbb{R}^+$ be some function. We define the* expected fitness operator $\mathcal{E}_f : \Lambda^X \to \mathbb{R}^+$ *as follows:*

$$\mathcal{E}_f(p) = \sum_{x \in X} f(x)p(x)$$

The expected fitness operator will be useful in defining the theme fitness function later on. It can also be used to express the selection operator more compactly as follows.

Remark 3. The selection operator can be expressed in terms of the Expected Fitness Operator as follows:

$$(\mathcal{S}_f p)(x) = \frac{f(x)p(x)}{\mathcal{E}_f(p)}$$

 ©IICAI

Definition 10. (Transmission Operator[3]) *Let X be a set, and let $M \in \Lambda_1^X$ be a transmission function over X. We define the transmission operator $\mathcal{T}_M : \Lambda^X \to \Lambda^X$ as follows:*

$$(\mathcal{T}_M p)(x) = \sum_{x' \in X} M(x|x')p(x')$$

The transmission operator will be used to model the effect of mutation on the genotypes that are selected as parents in each generation of a bfpEA

Definition 11. (Evolution Epoch Operator) *Let $B = (G, P, \phi, T)$ be a transmap, and let $f : P \to \mathbb{R}^+$ be some function. We define the evolution epoch operator $\mathcal{G}_{(B,f)} : \Lambda^G \to \Lambda^G$ as follows:*

$$\mathcal{G}_{(B,f)}(p) = \mathcal{T}_M \circ \mathcal{S}_{f \circ \phi} p$$

Given some bfpEA, an evolution epoch operator that is parameterized by the bfpEA's representation and fitness function models the advancement by one generation of a population of genotypes in the bfpEA. In section 6 we will see that when the mutation operator of a bfpEA is theme preserving, this operator can be used to express the advancement of a sub-population of genotypes that share the same theme.

The following definition associates a sequence of genotypic distributions with an EM using the operators we defined above. This sequence is a model of the generations of genotypic populations that are generated by a bfpEA.

Definition 12. (Genotype Distribution Sequence of an EM). *Let $E = (B, f, s)$ be some evolution machine. The genotype distribution sequence of E is a sequence $\{p^t\}_t$ of elements in Λ^G s.t. $p^0 = s$ and for any $t \in \mathbb{N}$:*

$$p^{t+1} = \mathcal{G}_{(B,f)} p^t$$

5.1 (β, ω)-preserving EMs

For some β and some ω, if the transmap of an EM E is (β, ω)-preserving then we say that E is (β, ω)-preserving. Consider a bfpEA Q, such that mutation of any genotype in Q is β-preserving with probability ω and β-altering with probability $(1 - \omega)$. Clearly, Q can be modelled by a (β, ω)-preserving EM. Hence we call Q a (β, ω)-preserving bfpEA.

6 Analysis of a (β, ω)-preserving EM

The following definition recalls the projection operator described in [9] and [8]. A projection operator projects a distribution over the domain of some function to the range of that function. The projection function is typically used to project distributions over the genotype set to the phenotype set. Here we will also use it to project genotype distributions onto theme sets.

[3] also called the Mixing Operator in [9] and [8]

 ©IICAI

Definition 13. (Projection Operator) *Let X, Y be some sets and let $\gamma : X \to Y$ be a function. We define the projection operator, $\Xi_\gamma : \Lambda^X \to \Lambda^Y$ as follows:*

$$(\Xi_\gamma p\ )(y) = \sum_{x \in \langle y \rangle} p(x)$$

We call $\Xi_\gamma p$ the γ-projection of p. To see that the range of Ξ_γ is indeed Λ^Y, i.e. that a projected distribution is also a distribution, note that for any $p \in \Lambda^X$,

$$\sum_{y \in Y} \sum_{x \in \langle y \rangle} p(x) = \sum_{x \in X} p(x) = 1$$

Given a countable set X, some set Y, a distribution p_X over X and some function-map $\gamma : X \to Y$, then for any element $y \in Y$ such that $(\Xi_\gamma p)(y) > 0$ we can define a new distribution over $\langle y \rangle$ by normalizing the probability mass of elements in $\langle y \rangle$ by the sum of their probability masses. We call this new distribution the γ-conditional distribution of p given k. Formally,

Definition 14. (Projection Conditional Distribution) *Let X, Y be sets and let $\gamma : X \to Y$ be a function. Let $p \in \Lambda^X$ be some distribution. For any $y \in Y$ such that $(\Xi_\gamma p)(y) > 0$, we define the γ-conditional distribution of p given y, to be a distribution $q \in \Lambda^{\langle y \rangle}$ s.t.*

$$q(x) = \frac{p(x)}{(\Xi_\gamma p)(y)}$$

The following definition describes a function that aggregates the fitness of all instances of a given theme in some generation.

Definition 15. (Theme Fitness Function) *Let G, K be sets, let $\beta : G \to K$ be a function, let E be an evolution machine with genotype set G, and let $\{p_G^t\}_{t \geq 0}$ be the genotype distribution sequence of E. For any $t \in \mathbb{Z}_0^+$, let p_K^t be the β-projection of p_G^t, and for all $k \in K$ such that $p_K^t(k) > 0$, let $p_{\langle k \rangle}^t$ be the projection conditional distribution of p_G^t given k. Then the β-theme fitness function of E at step t, ${}^\beta\mathfrak{F}_E^t : K \to \mathbb{R}^+$ is as follows:*

$${}^\beta\mathfrak{F}_E^t(k) = \begin{cases} \mathcal{E}_{f \circ \phi|_{\langle k \rangle}}({}^\omega p_{\langle k \rangle}^t) & \text{if } {}^\omega p_K^t(k) > 0 \\ 0 & \text{otherwise} \end{cases}$$

An important lemma is as follows:

Lemma 1 (Theme Preservation Lemma). *For all $\omega \in [0,1]$, let $B^\omega = (G, P, \phi, M)$ be a transmap such that for some set K and some $\beta : G \to K$, B^ω is (β,ω)-preserving. Let $E^\omega = (B^\omega, f, s)$ be a (β,ω)-preserving EM, and let $\{{}^\omega p_G^t\}_{t \geq 0}$ be the genotype distribution sequence of E^ω. For all $t \in \mathbb{Z}_0^+$, let ${}^\omega p_K^t$*

 ©IICAI

be the β-projection of ${}^{\omega}p^t_G$. For all $k \in K$ such that ${}^{\omega}p^t_K(k) > 0$, let ${}^{\omega}p^t_{\langle k \rangle}$ be the β-conditional distribution of ${}^{\omega}p^t_G$ given k. Then, for all $t \in \mathbb{Z}^+_0$:

$$D(\mathcal{S}_{\beta\mathfrak{F}^t_{E^\omega}}({}^{\omega}p^t_K) \,||\, {}^{\omega}p^{t+1}_K) \leq \log \frac{1}{\omega}$$

and for all $k \in K$ such that ${}^{\omega}p^t_K(k) > 0$,

$$D(\mathcal{G}_{(B^1|_{\langle k \rangle}, f)}({}^{\omega}p^t_{\langle k \rangle}) \,||\, {}^{\omega}p^{t+1}_{\langle k \rangle}) \leq \log\left(1 + \frac{1-\omega}{\omega(\mathcal{S}_{\beta\mathfrak{F}^t_{\mathcal{E}^\omega}}\, {}^{\omega}p^t_K)(k)}\right)$$

where D is the Kullback-Liebler Divergence[4]*.*

The central result of this paper follows from this lemma by treating the right-hand-sides of both inequalities as functions of ω and by observing that these functions are continuous over the interval (0,1].

Theorem 1 (Theme Preservation Theorem). *For all $t \in \mathbb{Z}^+_0$, as $\omega \to 1$,*

$${}^{\omega}p^{t+1}_K \to \mathcal{S}_{\beta\mathfrak{F}^t_{E^\omega}}({}^{\omega}p^t_K) \tag{1}$$

and for all $k \in K$ such that ${}^{\omega}p^t_K(k) > 0$,

$${}^{\omega}p^{t+1}_{\langle k \rangle} \to \mathcal{G}_{(B^1|_{\langle k \rangle}, f)}({}^{\omega}p^t_{\langle k \rangle}) \tag{2}$$

with equality when $\omega = 1$

Before we begin the proof of the lemma, let us understand the implications of the theme preservation theorem. When $\omega = 1$, for any $k \in K$ such that ${}^{1}p^0_K(k) > 0$, $\{\, {}^{1}p^t_{\langle k \rangle}\}_{t \geq 0}$ is the genotype distribution sequence of the EM $E^1_k = (B^1|_{\langle k \rangle}, f,\, {}^{1}p^0_{\langle k \rangle})$. Let us call such an EM a (β, k)-subEM of E^1, let us call the genotype distribution sequence of E^1 the evolutionary process of E^1 and for any $k \in K$, let us call the genotype distribution sequence of E^1_k the evolutionary thread of E^1_k. Then, equation (2) shows that evolutionary process of E^1 can be decomposed, or "factored", into the evolutionary threads of subEMs of E^1. For any $k \in K$ such that ${}^{1}p^t_K(k) > 0$, the (β, k)-subEM uses the (β, k)-subtransmap of B^1. Thus there is a correspondence between instantiated themes and evolutionary threads. Each thread can be thought to evolve the non-thematic parts of all genotypes

[4] For any two distributions p and q over the same domain, the Kullback-Liebler divergence $D(p \,||\, q)$ is an asymmetric measure of the "distance" between p and q. It has the property that $D(p \,||\, q) = 0 \Rightarrow p = q$. More technically (and less relevantly), it is a measure of the inefficiency of assuming that a distribution over some domain is q when in fact it is p. Given an optimal code for p, the lower bound of the average number of bits per symbol needed to communicate symbols drawn from p is $H(p)$. If however an optimal code for q is used for communication then the lower bound of the average number of bits per symbol needed to send symbols drawn from the distribution p is $H(p) + D(p \,||\, q)$ [3]

 ©IICAI

which share some common theme. For these reasons we will call equation (2) the *theme-thread correspondence equation*.

Observe that by the definition of ${}^{1}p^{t}_{\langle k \rangle}$ for all t in the premise of the theorem, we have that in any generation t the "weighting" of the population of any sub-EM E^1_k at generation t within the overall population of E^1 is determined by the probability mass of k given by ${}^{1}p^{t}_{K}(k)$. The sequence $\{\, {}^{1}p^{t}_{K} \}_{t \in \mathbb{Z}^+_0}$ proceeds according to equation (1). Unpacking this equation we see that the probability mass of the themes in some generation $t+1$, given by ${}^{1}p^{t+1}_{K}$, is determined by the application of the selection operator to the distribution ${}^{1}p^{t}_{K}$. The fitness function employed by the selection operator is the theme fitness function ${}^{\beta}\mathfrak{F}^{t}_{E^1}$ defined earlier. We call equation (1) the *theme selection equation* because it shows that when $\omega = 1$ ordinary selection of the genotypes implicitly implements selection over the themes. The only way that the evolutionary threads interact is by the transfer of theme probability mass between threads in each generation as described by the theme selection equation.

When $\omega \approx 1$ the theme preservation corollary shows that the theme selection equation and the theme-thread correspondence equation hold approximately in each generation. We will explore the consequences of this in section 7. We now prove the lemma.

Proof. The proof of the first inequality rests on two claims.

Claim 1 *For all* $\omega \in [0,1], t \in \mathbb{Z}^+_0$,

$$\sum_{g \in \langle k \rangle} (\mathcal{T}_{M_P} \mathcal{S}_{f \circ \phi}\, {}^{\omega}p^{t}_{G})(g) = (\mathcal{S}_{\beta \mathfrak{F}^{t}_{E^{\omega}}}\, {}^{\omega}p^{t}_{K})(k)$$

Claim 2 *let* $u, q, r \in \Lambda^K$ *be distributions such that* $u(k) = \omega q(k) + (1-\omega) r(k)$. *Then* $D(q||u) \leq \log \frac{1}{\omega}$

We assume these claims for now and prove them later. For all $\omega \in [0,1], t \in \mathbb{Z}^+_0$, see that

$$\begin{aligned}
{}^{\omega}p^{t+1}_{K}(k) &= \sum_{g \in \langle k \rangle} {}^{\omega}p^{t+1}_{G}(g) \\
&= \omega \sum_{g \in \langle k \rangle} \mathcal{T}_{M_P} \mathcal{S}_{f \circ g}\, {}^{\omega}p^{t+1}_{G}(g) + (1-\omega) \sum_{g \in \langle k \rangle} \mathcal{T}_{M_A} \mathcal{S}_{f \circ \phi}\, {}^{\omega}p^{t+1}_{G}(g) \\
&= \omega \mathcal{S}_{\beta \mathfrak{F}^{t}_{E^{\omega}}}\, {}^{\omega}p^{t}_{K}(k) + (1-\omega) \sum_{g \in \langle k \rangle} \mathcal{T}_{M_A} \mathcal{S}_{f \circ \phi}\, {}^{\omega}p^{t+1}_{G}(g)
\end{aligned}$$

Where the last equality follows by claim 1. Substituting $(\mathcal{S}_{\beta \mathfrak{F}^{t}_{E^{\omega}}}\, {}^{\omega}p^{t}_{K})(k)$ for $q(k)$ and $\sum_{g \in \langle k \rangle} \mathcal{T}_{M_A}(\mathcal{S}_{f \circ \phi}\, {}^{\omega}p^{t+1}_{G})(g)$ for $r(k)$ in claim 2, see that $u(k) = {}^{\omega}p^{t+1}_{K}(k)$. So, $D(\mathcal{S}_{\beta \mathfrak{F}^{t}_{E^{\omega}}}\, {}^{\omega}p^{t}_{K} || \, {}^{\omega}p^{t+}_{K}) \leq \log \frac{1}{\omega}$.

The proof of the second inequality of the lemma rests on two further claims.

Claim 3 *For all* $\omega \in [0,1]$, $t \in \mathbb{Z}^+_0$, $k \in K$ *such that* $p^{t}_{K}(k) > 0$,

$$\frac{(\mathcal{T}_{M_P} \mathcal{S}_{f \circ \phi}\, {}^{\omega}p^{t}_{G})(g)}{\sum_{g' \in \langle k \rangle} (\mathcal{T}_{M_P} \mathcal{S}_{f \circ \phi}\, {}^{\omega}p^{t}_{G})(g')} = (\mathcal{G}_{(B^1|_{\langle k \rangle}, f)}\, {}^{\omega}p^{t}_{\langle k \rangle})(g)$$

 ©IICAI

Claim 4 *Let $q, r \in \Lambda^G$ be distributions. For any $k \in K$ such that $(\Xi_\beta q)(k) > 0$, let $u, v \in \Lambda^{\langle k \rangle}$ be defined as follows:*

$$u(g) = \frac{\omega q(g) + (1-\omega) r(g)}{\sum_{g' \in \langle k \rangle} \omega q(g') + (1-\omega) r(g') + (1-\omega) r(g')}$$

$$v(g) = \frac{q(g)}{\sum_{g' \in \langle k \rangle} q(g')}.$$

then,

$$D(v||u) \leq \log \quad 1 + \frac{1-\omega}{\omega \sum_{g' \in \langle k \rangle} q(g')} \quad .$$

Once again we assume these claims for now and prove them later. For all $\omega \in [0,1]$, $t \in \mathbb{Z}_0^+$, $k \in K$ such that ${}^{\omega}p_K^t(k) > 0$,

$$\begin{aligned} {}^{\omega}p_{\langle k \rangle}^{t+1}(g) =& \frac{\omega p_G^{t+1}(g)}{\sum_{g' \in \langle k \rangle} {}^{\omega}p_G^{t+1}(g')} \\ =& \frac{\omega \mathcal{T}_{M_P} \mathcal{S}_{f \circ \phi} \, {}^{\omega}p_G^t(g) + (1-\omega) \mathcal{T}_{M_A} \mathcal{S}_{f \circ \phi} \, {}^{\omega}p_G^t(g)}{\sum_{g \in \langle k \rangle} \omega \mathcal{T}_{M_P} \mathcal{S}_{f \circ \phi} \, {}^{\omega}p_G^t(g') + (1-\omega) \mathcal{T}_{M_A} \mathcal{S}_{f \circ \phi} \, {}^{\omega}p_G^t(g')} \end{aligned}$$

Substituting $(\mathcal{T}_{M_P} \mathcal{S}_{f \circ \phi} \, {}^{\omega}p_G^t)$ for q and $(\mathcal{T}_{M_A} \mathcal{S}_{f \circ \phi} \, {}^{\omega}p_G^t)$ for r in claim 4, see that $u = {}^{\omega}p_{\langle k \rangle}^{t+1}$ and by claim 3, $v = \mathcal{G}_{(B^1|_{\langle k \rangle}, f)} \, {}^{\omega}p_{\langle k \rangle}^t(g)$. So,

$$\begin{aligned} D(\mathcal{G}_{(B^1|_{\langle k \rangle}, f)} \, {}^{\omega}p_{\langle k \rangle}^t || \, {}^{\omega}p_{\langle k \rangle}^{t+1}) &= \log \quad 1 + \frac{1-\omega}{\omega \sum_{g' \in \langle k \rangle} \mathcal{T}_{M_P} \mathcal{S}_{f \circ \phi} \, {}^{\omega}p_G^t(g)} \\ &= \log \quad 1 + \frac{1-\omega}{\omega (\mathcal{S}_{\beta_{\mathfrak{F}_{E^\omega}^t}} \, {}^{\omega}p_K^t)(k)} \end{aligned}$$

where the last equality follows by claim 1.
Proof of Claim 1:

$$\begin{aligned} \sum_{g \in \langle k \rangle} \mathcal{T}_{M_P} \mathcal{S}_{f \circ \phi} \, {}^{\omega}p_G^t(g) &= \sum_{g \in \langle k \rangle} \sum_{g' \in G} M_P(g|g') (\mathcal{S}_{f \circ g} \, {}^{\omega}p_G^t)(g') \\ &= \sum_{g' \in G} (\mathcal{S}_{f \circ \phi} \, {}^{\omega}p_G^t)(g') \sum_{g \in \langle k \rangle} M_P(g|g') \end{aligned}$$

For any $k \in K$ any $g \in \langle k \rangle$, and any $g' \notin \langle k \rangle$, by definition of a Preserving Transmission Function in (def 2), $M_P(g|g') = 0$. So,

$$\sum_{g \in \langle k \rangle} \mathcal{T}_{M_P} \mathcal{S}_{f \circ \phi} \, {}^{\omega}p_G^t(g) = \sum_{g' \in \langle k \rangle} (\mathcal{S}_{f \circ \phi} \, {}^{\omega}p_G^t)(g') \sum_{g \in \langle k \rangle} M_P(g|g')$$

By proposition 1,

$$\sum_{g \in \langle k \rangle} \mathcal{T}_{M_P} \mathcal{S}_{f \circ \phi} \, {}^{\omega}p_G^t(g) = \sum_{g' \in \langle k \rangle} (\mathcal{S}_{f \circ \phi} \, {}^{\omega}p_G^t)(g')$$

 ©IICAI

By definition of the Selection Operator in terms of the Expected Fitness Operator in (remark 3),

$$\underset{g\in\langle k\rangle}{\mathcal{T}_{M_P}\mathcal{S}_{f\circ\phi}}\ {}^{\omega}p_G^t(g) = \underset{g'\in\langle k\rangle}{\frac{f\circ\phi(g')\,{}^{\omega}p_G^t(g')}{\mathcal{E}_{f\circ\phi}({}^{\omega}p_G^t)}} \tag{3}$$

We examine the following two cases,

case i: k such that ${}^{\omega}p_K^t(k) = 0$. This implies that for all $g \in \langle k\rangle$, ${}^{\omega}p_G^t(g) = 0$, so using equation (3), $\underset{g\in\langle k\rangle}{T_{M_P}\mathcal{S}_{f\circ\phi}}\ {}^{\omega}p_G^t(g) = 0$.

case ii: k such that ${}^{\omega}p_K^t(k) > 0$. Using the definition of ${}^{\omega}p_{\langle\cdot\rangle}^t$ in the numerator

$$\begin{aligned}\underset{g\in\langle k\rangle}{\mathcal{T}_{M_P}\mathcal{S}_{f\circ\phi}}\ {}^{\omega}p_G^t(g) &= \underset{g'\in\langle k\rangle}{\frac{f\circ\phi(g')\,{}^{\omega}p_K^t(k)\,{}^{\omega}p_{\langle k\rangle}^t(g')}{\mathcal{E}_{f\circ\phi}({}^{\omega}p_G^t)}}\\ &= \frac{{}^{\omega}p_K^t(k)\quad\underset{g'\in\langle k\rangle}{f\circ\phi(g')\,{}^{\omega}p_{\langle k\rangle}^t(g')}}{\mathcal{E}_{f\circ\phi}({}^{\omega}p_G^t)}\\ &= \frac{{}^{\omega}p_K^t(k)\quad\underset{g'\in\langle k\rangle}{f\circ\phi|_{\langle k\rangle}(g')\,{}^{\omega}p_{\langle k\rangle}^t(g')}}{\mathcal{E}_{f\circ\phi}({}^{\omega}p_G^t)}\end{aligned}$$

where the last equation follows from the definition of restriction. Using the Expected Fitness Operator defined in (def 9) to express the numerator,

$$\underset{g\in\langle k\rangle}{\mathcal{T}_{M_P}\mathcal{S}_{f\circ\phi}}\ {}^{\omega}p_G^t(g) = \frac{{}^{\omega}p_K^t(k)\mathcal{E}_{f\circ\phi|_{\langle k\rangle}}({}^{\omega}p_{\langle k\rangle}^t)}{\mathcal{E}_{f\circ\phi}({}^{\omega}p_G^t)} \tag{4}$$

By expansion of the Expected Fitness Operator in the denominator using (def 9),

$$\underset{g\in\langle k\rangle}{\mathcal{T}_{M_P}\mathcal{S}_{f\circ\phi}}\ {}^{\omega}p_G^t(g) = \frac{{}^{\omega}p_K^t(k)\mathcal{E}_{f\circ\phi|_{\langle k\rangle}}({}^{\omega}p_{\langle k\rangle}^t)}{\underset{g'\in G}{f\circ\phi(g')\,{}^{\omega}p_G^t(g')}}$$

Using the definition of ${}^{\omega}p_{\langle\cdot\rangle}^t$ in the denominator,

$$\begin{aligned}\underset{g\in\langle k\rangle}{\mathcal{T}_{M_P}\mathcal{S}_{f\circ\phi}}\ {}^{\omega}p_G^t(g) &= \frac{{}^{\omega}p_K^t(k)\mathcal{E}_{f\circ\phi|_{\langle k\rangle}}({}^{\omega}p_{\langle k\rangle}^t)}{\underset{k'\in K}{\quad\underset{g'\in\langle k'\rangle}{f\circ\phi(g')\,{}^{\omega}p_K^t(k')\,{}^{\omega}p_{\langle k'\rangle}^t(g')}}}\\ &= \frac{{}^{\omega}p_K^t(k)\mathcal{E}_{f\circ\phi|_{\langle k\rangle}}({}^{\omega}p_{\langle k\rangle}^t)}{\underset{k'\in K}{{}^{\omega}p_K^t(k')\quad\underset{g'\in\langle k'\rangle}{f\circ\phi(g')\,{}^{\omega}p_{\langle k'\rangle}^t(g')}}}\end{aligned}$$

Using the Expected Fitness Operator defined in (def 9) to express the denominator,

$$\underset{g\in\langle k\rangle}{\mathcal{T}_{M_P}\mathcal{S}_{f\circ\phi}}\ {}^{\omega}p_G^t(g) = \frac{{}^{\omega}p_K^t(k)\mathcal{E}_{f\circ\phi|_{\langle k\rangle}}({}^{\omega}p_{\langle k\rangle}^t)}{\underset{k'\in K}{{}^{\omega}p_K^t(k')\mathcal{E}_{f\circ\phi|_{\langle k'\rangle}}({}^{\omega}p_{\langle k'\rangle}^t(g'))}}$$

Hence for all $k \in K$, using the definition of the Theme Fitness Function in (def 15),

$$\underset{g\in\langle k\rangle}{\mathcal{T}_{M_P}\mathcal{S}_{f\circ\phi}}\ {}^{\omega}p_G^t(g) = \frac{p_K^t(k)\,{}^{\beta}\mathfrak{F}_{E^\omega}^t(k)}{\underset{k'\in K}{{}^{\omega}p_K^t(k')\,{}^{\beta}\mathfrak{F}_{E^\omega}^t(k')}}$$

 ©IICAI

Using the definition of the Selection Operator (def 8), we obtain the result in claim 1.

$$\underset{g\in\langle k\rangle}{\mathcal{T}_{M_P}\mathcal{S}_{f\circ\phi}\,{}^{\omega}p_G^t(g)} = \mathcal{S}_{\beta_{\mathfrak{F}^t_{E^\omega}}}({}^{\omega}p_K^t)$$

Proof of Claim 3: By the definition of the Transmission operator in (def 10), for all $k \in K$ such that ${}^{\omega}p_K^t(k) > 0$, and for all $g \in \langle k\rangle$,

$$\frac{(\mathcal{T}_{M_P}\mathcal{S}_{f\circ\phi}\,{}^{\omega}p_G^t)(g)}{\underset{g'\in\langle k\rangle}{(\mathcal{T}_{M_P}\mathcal{S}_{f\circ\phi}\,{}^{\omega}p_G^t)(g')}} = \underset{g'\in G}{M_P(g|g')\mathcal{S}_{f\circ g}\,{}^{\omega}p_G^t(g')} \cdot \frac{1}{\underset{g'\in\langle k\rangle}{(\mathcal{T}_{M_P}\mathcal{S}_{f\circ\phi}\,{}^{\omega}p_G^t)(g')}}$$

Using the alternate form of the Selection Operator given in (remark 3) and using (4) we have that,

$$\frac{(\mathcal{T}_{M_P}\mathcal{S}_{f\circ\phi}\,{}^{\omega}p_G^t)(g)}{\underset{g'\in\langle k\rangle}{(\mathcal{T}_{M_P}\mathcal{S}_{f\circ\phi}\,{}^{\omega}p_G^t)(g')}} = \frac{\sum_{g'\in G} M_P(g|g')f\circ\phi(g)\,{}^{\omega}p_G^t(g')}{\mathcal{E}_{f\circ\phi}({}^{\omega}p_G^t)} \cdot \frac{\mathcal{E}_{f\circ\phi}({}^{\omega}p_G^t)}{{}^{\omega}p_K^t(k)\mathcal{E}_{f\circ\phi|_{\langle k\rangle}}({}^{\omega}p_{\langle k\rangle}^t)}$$

Cancelling terms and consolidating yields

$$\frac{(\mathcal{T}_{M_P}\mathcal{S}_{f\circ\phi}\,{}^{\omega}p_G^t)(g)}{\underset{g'\in\langle k\rangle}{(\mathcal{T}_{M_P}\mathcal{S}_{f\circ\phi}\,{}^{\omega}p_G^t)(g')}} = \frac{\sum_{g'\in G} M_P(g|g')f\circ\phi(g')\frac{{}^{\omega}p_G^t(g')}{{}^{\omega}p_K^t(k)}}{\mathcal{E}_{f\circ\phi|_{\langle k\rangle}}\,{}^{\omega}p_{\langle k\rangle}^t}$$

Note that for any $g \in \langle k\rangle$ and any $g' \notin \langle k\rangle$, $M(g|g') = 0$. Also note that $\frac{{}^{\omega}p_G^t(g)}{{}^{\omega}p_K^t(k)} = {}^{\omega}p_{\langle k\rangle}^t(g)$. Hence,

$$\frac{(\mathcal{T}_{M_P}\mathcal{S}_{f\circ\phi}\,{}^{\omega}p_G^t)(g)}{\underset{g'\in\langle k\rangle}{(\mathcal{T}_{M_P}\mathcal{S}_{f\circ\phi}\,{}^{\omega}p_G^t)(g')}} = \frac{\sum_{g'\in\langle k\rangle} M_P(g|g')f\circ\phi(g')\,{}^{\omega}p_{\langle k\rangle}^t(g')}{\mathcal{E}_{f\circ\phi|_{\langle k\rangle}}\,{}^{\omega}p_{\langle k\rangle}^t}$$

By the definition of Restriction of a function,

$$\frac{(\mathcal{T}_{M_P}\mathcal{S}_{f\circ\phi}\,{}^{\omega}p_G^t)(g)}{\underset{g'\in\langle k\rangle}{(\mathcal{T}_{M_P}\mathcal{S}_{f\circ\phi}\,{}^{\omega}p_G^t)(g')}} = \frac{\sum_{g'\in\langle k\rangle} M_P|_{\langle k\rangle}(g|g')f\circ\phi|_{\langle k\rangle}(g')\,{}^{\omega}p_{\langle k\rangle}^t(g')}{\mathcal{E}_{f\circ\phi|_{\langle k\rangle}}\,{}^{\omega}p_{\langle k\rangle}^t}$$

Using the alternate form of the Selection Operator given in (remark 3), the definition of the Transmission Operator in (def 10), we get

$$\begin{aligned}\frac{(\mathcal{T}_{M_P}\mathcal{S}_{f\circ\phi}\,{}^{\omega}p_G^t)(g)}{\underset{g'\in\langle k\rangle}{(\mathcal{T}_{M_P}\mathcal{S}_{f\circ\phi}\,{}^{\omega}p_G^t)(g')}} &= \underset{g\in\langle k\rangle}{M_P|_{\langle k\rangle}(g|g')(S_{f\circ\phi|_{\langle k\rangle}}\,{}^{\omega}p_{\langle k\rangle}^t)(g')} \\ &= (\mathcal{T}_{M_P|_{\langle k\rangle}}\mathcal{S}_{f\circ\phi|_{\langle k\rangle}}p_{\langle k\rangle}^t)(g)\end{aligned}$$

As M_P is theme preserving, by (proposition 2), $(\langle k\rangle, P, M_P|_{\langle k\rangle}, \phi|_{\langle k\rangle})$ is a transmap. Hence, using the definition of the Evolution Epoch Operator in (def 11),

$$\frac{(\mathcal{T}_{M_P}\mathcal{S}_{f\circ\phi}\,{}^{\omega}p_G^t)(g)}{\underset{g'\in\langle k\rangle}{(\mathcal{T}_{M_P}\mathcal{S}_{f\circ\phi}\,{}^{\omega}p_G^t)(g')}} = (\mathcal{G}_{((\langle k\rangle,P,M_P|_{\langle k\rangle},\phi|_{\langle k\rangle}),f)}p_{\langle k\rangle}^t)(g)$$

 ©IICAI

But $(\langle k\rangle, P, M_P|_{\langle k\rangle}, \phi|_{\langle k\rangle}) = (G, P, M_P, \phi)|_{\langle k\rangle} = B^1|_{\langle k\rangle}$. We thus obtain the result in claim 3.

Proof of Claim 2:

$$\begin{aligned} D(q||u) &= \sum_{k\in K} q(k) \log \frac{q(k)}{\omega q(k) + (1-\omega) r(k)} \\ &\leq \sum_{k\in K} q(k) \log \frac{q(k)}{\omega q(k)} \\ &= \log \frac{1}{\omega} \cdot \sum_{k\in K} q(k) \\ &= \log \frac{1}{\omega} \end{aligned}$$

Proof of Claim 4:

$$\begin{aligned} D(v||u) &= \sum_{g\in\langle k\rangle} \frac{q(g)}{\sum_{g'\in\langle k\rangle} q(g')} \log \frac{\sum_{g'\in\langle k\rangle} q(g')}{\sum_{g'\in\langle k\rangle} \omega q(g') + (1-\omega) r(g')} \cdot \frac{\sum_{g'\in\langle k\rangle} \omega q(g') + (1-\omega) r(g')}{\omega q(g) + (1-\omega) r(g)} \\ &\leq \sum_{g\in\langle k\rangle} \frac{q(g)}{\sum_{g'\in\langle k\rangle} q(g')} \log \frac{\sum_{g'\in\langle k\rangle} q(g')}{\sum_{g'\in\langle k\rangle} \omega q(g') + (1-\omega) r(g')} \cdot \frac{\sum_{g'\in\langle k\rangle} \omega q(g') + (1-\omega) r(g')}{\omega q(g)} \\ &= \sum_{g\in\langle k\rangle} \frac{q(g)}{\sum_{g'\in\langle k\rangle} q(g')} \log \frac{\sum_{g'\in\langle k\rangle} \omega q(g') + (1-\omega) r(g')}{\omega \sum_{g'\in\langle k\rangle} q(g')} \end{aligned}$$

See that $\sum_{g'\in\langle k\rangle} r(g') \leq 1$. So,

$$\begin{aligned} D(v||u) &\leq \sum_{g\in\langle k\rangle} \frac{q(g)}{\sum_{g'\in\langle k\rangle} q(g')} \log \frac{\sum_{g'\in\langle k\rangle} \omega q(g') + (1-\omega)}{\omega \sum_{g'\in\langle k\rangle} q(g')} \\ &= \log \left(1 + \frac{1-\omega}{\omega \sum_{g'\in\langle k\rangle} q(g')}\right)^{\frac{\sum_{g\in\langle k\rangle} q(g)}{\sum_{g'\in\langle k\rangle} q(g')}} \\ &= \log \left(1 + \frac{1-\omega}{\omega \sum_{g'\in\langle k\rangle} q(g')}\right) \end{aligned}$$

□

7 Discussion

In the previous section we analyzed the behavior of a (β, ω)-preserving EM which is a model for a (β, ω)-preserving bfpEA. In this section we interpret these theoretical results to infer qualitative and quantitative aspects about the behavior of a (β, ω)-preserving bfpEA. Let Q^ω be a (β, ω)-preserving bfpEA and let E^ω be a (β, ω)-preserving EM that models Q^ω. We will first focus on the behavior of Q^ω when $\omega = 1$. This behavior is shown schematically in (fig. 1).

 ©IICAI

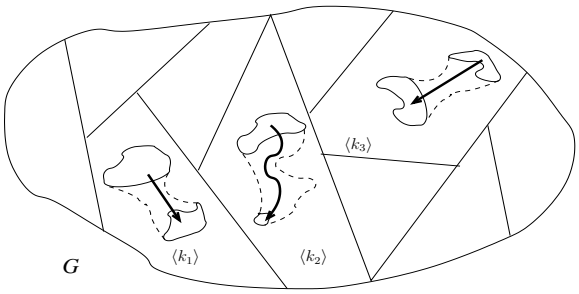


Fig. 1. Schematic depiction of the behavior of a $(\beta, 1)$-preserving bfpEA

The function β partitions the genotype set into theme classes. A shaded region within some theme class $\langle k \rangle$ depicts a subpopulation of k-themed genotypes at some time t [5] The initial population of Q^1 is comprised of the union of subpopulations in different theme classes. Some themes may not be present at all in the initial population of Q^1. This is depicted by the empty theme classes in the figure. The sizes of the subpopulations of Q^1 vary from generation to generation as evolution proceeds. This is depicted as a change in the sizes of the shaded regions depicting subpopulations. Let n be the fixed total population size of Q^1. For any $t \in \mathbb{Z}_0^+$, let F_k^t be the total fitness of all the individuals in some k-themed subpopulation at time t. Then by the theme selection equation the size of the k-themed sub-population in generation $t+1$, denoted by n_k^{t+1}, is given by $n_k^{t+1} \approx n \frac{F_k^t}{\sum_{k' \in K} F_{k'}^t}$.

From the theme-thread correspondence equation we can infer that apart from the change in the size of the subpopulations in each generation (as described above), the evolution of any k-themed subpopulation proceeds independently of the evolution of other subpopulations. If R is the representation that is used by Q^1, then we can think of each k-themed subpopulation as evolving within a separate variable-population-size bfpEA that uses a (β, k)-subrepresentation of R; let us call this a (β, k)-sub-bfpEA of Q^1.

For any theme k and any generation t, the value F_k^t can be thought of as the fitness of the (β, k)-sub-bfpEA in generation t. In each generation $t \in \mathbb{Z}_0^+$, the population size of the (β, k)-sub-bfpEA is given by n_k^t. If one thinks of the population size of a sub-bfpEA in some generation as the amount of search resources allocated to its representation in that generation, then in each generation Q^1 reallocates its search resources amongst the subrepresentations of its sub-bfpEAs in proportion to the fitness of the populations of the sub-bfpEAs. We call this behavior *Subrepresentation selecting multithreaded evolution* (SSME).

[5] When interpreting this figure and (fig. 2) the reader should bear in mind that while subpopulations are depicted as sets they are really multi-sets. We nevertheless use the size of a shaded area to depict the size of some sub-population of some theme class.

 ©IICAI

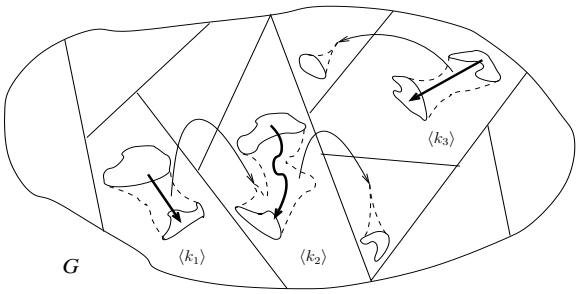


Fig. 2. Schematic depiction of the behavior of a (β, ω)-preserving bfpEA when $\omega \approx 1$

Note that when $\omega = 1$, if some theme k' is not instantiated by some genotype of the initial generation of Q^ω, then k' will never be present in any of the genotypes in any subsequent generations. Thus the (β, k')-subrepresentation of Q^ω will never be explored. So, in each generation, Q^ω will only perform subrepresentation selection on the subrepresentations that are already present in the initial generation.

When $\omega \approx 1$ we say that the bfpEA Q^ω is *mostly-theme-preserving*. By the corollary to theorem 1, the subtransmap selecting multithreaded evolution equations approximately hold for E^ω. Therefore, we infer that Q^ω *approximately* implements subrepresentation selecting multithreaded evolution. The key qualitative difference between the SSME behavior of Q^1 and the behavior of Q^ω is the result of the small number of theme altering mutations in each generation of Q^ω. Figuratively speaking, the child genotypes produced by such mutation "jump over" the theme class partitions as shown in (fig. 2) and land in theme classes that are different from those of their parents. Thus in each generation of Q^ω new themes may be instantiated which were not present in previous generations. When $\omega \approx 1$ themes correspond approximately to the subrepresentations used by sub-bfpEAs, so another way of saying the above is that in each generation sub-bfpEAs with new subrepresentations may be generated by Q^ω. As Q^ω approximately implements SSME, these new subrepresentations will be subject to subrepresentation selection in subsequent generations. Since Q^ω generates new subrepresentations in addition to approximately performing SSME, we call its behavior *Subrepresentation evolving multithreaded evolution* (SEME). Thus a (β, ω)-preserving bfpEA performs second-order search over the a space of subrepresentations and in each generation exploits promising subrepresentations for first-order evolutionary search.

8 A Comparison With Schema Theory

We now compare the analytical method and results in this paper with the methods and results found in other works on schema theory. Schema theorems [4, 5,

 ©IICAI

7] define specific subsets of the genotype space and model the flow of individuals of an evolving population between these subsets. The subsets are defined using similarity templates called schemata. A genotype that matches a schema is said to "belong" to it.

When compared to the schema theorems, the theme preservation theorem makes different assumptions about the Evolutionary Algorithms being modelled:

1. Schema theorems typically assume that variation consists of mutation *and* recombination whereas the theme preservation theorem assumes that mutation is the only variation operator that is used.
2. The analysis in each schema theorem is specific to: i) a particular genotypic data structure (e.g. strings or trees), ii) a particular way of performing recombination (e.g. uniform crossover in the case of strings, homologous one point crossover in the case of trees) and mutation (e.g. Bitflips in strings with probability 1/l, where l is the length of a string) On the other hand the theme preservation theorem is *not* specific to i) a particular genotypic datastructure, or ii) a particular way of performing mutation (differences in the genotypic data structure and mutation operator are treated by using a different genotype set G and different mutation-only transmission function M)

Also, the way that subsets of the genotype space are defined and the properties they have differ between the schema theorems and the theme preservation theorem:

1. Schema theorems use similarity templates (e.g. $**101*1$ in the case of Holland's schema theorem) to define subsets of the genotype space, whereas the subsets in the theme preservation theorem are the equivalence classes induced by a mapping from the genotype set into a theme set.
2. A genotype may belong to more than one schema, i.e. the subsets corresponding to the schemata may overlap. On the other hand, given some theme map β, a genotype must belong to one and only one β-theme class, i.e. the subsets are disjoint. Furthermore, the theme preservation theorem requires that the probability that the mutant of a genotype will stay within the parent's subset is a constant ω for all subsets. And, for the theorem to have epistemological impact, ω must be close to 1.

Because of these differences, comparing the result of the theme preservation theorem with that of other schema theorems is a little like comparing apples with oranges, yet at a high level all these theorems ultimately say something about the flow of an evolving population between subsets of the genotype space, and it is on these grounds that we make the following comparison. We note that:

1. The theme selection equation which describes the inter-subset flow of the population in each evolutionary epoch is much more concise than a typical schema evolution equation. The conciseness of expression in this macroscopic level equation is achieved by using the same selection operator that is used in the microscopic level evolution equation (in definition 11). Therefore one's

 ©IICAI

familiarity with the effect of this operator in the microscopic level equation makes it easier to understand its effect in the macroscopic level equation.

2. The theme - thread correspondence equation gives us an intuitive way to understand what is going on *within* each subset. i.e. the *intra*-subset flows. It shows that when $\omega \approx 1$ an approximation of evolution occurs within each subset. Schema theorems do not give a similar pithy understanding of intra-subset flows.

Our intent in this section has been to illuminate our analytical method and results by comparing them with other methods and results. We do not wish to imply that one approach to analyzing EAs is better than another. As we've noted schema theorems analyze the behaviour of EAs which use a recombination operator. Thus existing work on schema theorems analyze algorithms that are more complicated than the ones considered in this work. Note however that our analysis of simpler systems, and the way in which we define the subsets that are analyzed, gives us results that are more intuitively understandable and are *not* specific to a particular data-structure.

Acknowledgements: We thank all the members of the DEMO lab, especially Anthony Bucci and Shivakumar Vishwanathan, for very valuable feedback on the content and form of this paper. We also thank the anonymous reviewers for their helpful comments.

References

1. Altenberg, L. The evolution of evolvability in genetic programming. In *Advances in Genetic Programming*, K. E. Kinnear, Jr., Ed. MIT Press, 1994, ch. 3, pp. 47–74.
2. Burjorjee, K. M., and Pollack, J. B. Theme preservation and the evolution of representation. In *Theory of Representation Workshop, Genetic and Evolutionary Computation Conference* (2005).
3. Cover, T. M., and Thomas, J. A. *Elements of Information Theory*. John Wiley & sons, 1991.
4. Holland, J. H. *Adaptation in Natural and Artificial Systems: An Introductory Analysis with Applications to Biology, Control, and Artificial Intelligence*. University of Michigan, 1975.
5. Langdon, W. B., and Poli, R. *Foundations of Genetic Programming*. Springer-Verlag, 2002.
6. Radcliffe, N. J. The algebra of genetic algorithms. *Annals of Mathematics and Artificial Intelligence 10* (1994), 339–384.
7. Stephens, C., and Waelbroeck, H. Schemata evolution and building blocks. *Evolutionary Computation 7*, 2 (1999), 109–124.
8. Toussaint, M. *The Evolution of Genetic Representations and Modular Neural Adaptation*. PhD thesis, Institut fr Neuroinformatik, Ruhr-Universit-Bochum, Germany, 2003.
9. Vose, M. D. *The simple genetic algorithm: foundations and theory*. MIT Press, Cambridge, MA, 1999.
10. Wagner, G. P., and Altenberg, L. Complex adaptations and the evolution of evolvability. *Evolution* (1996), In press.

 ©IICAI