What Basis for Genetic Dynamics?

Chryssomalis Chryssomalakos and Christopher R. Stephens

Instituto de Ciencias Nucleares UNAM
Circuito Exterior Ciudad Universitaria
Apdo. Postal 70-543 México D.F. 04510 MÉXICO
chryss,stephens@nuclecu.unam.mx

Abstract. We present a covariant form for genetic dynamics and show how different formulations are simply related by linear coordinate transformations. In particular, in the context of the simple genetic algorithm, we show how the Vose model, in either the string or Walsh bases, is related to recent coarse-grained formulations that are naturally interpreted in terms of the Building Block basis (BBB). We also show that the latter is dual to the Taylor basis. The tensor product structure of the dynamical equations is analyzed, permitting the factorization of the N-bit operators in 1-bit factors.

1 Introduction

There has been a recent trend [1–3] in evolutionary computation (EC) towards a unified point of view, wherein different branches of EC, such as genetic algorithms (GA's) and genetic programming (GP), have been given a common, rigorous, theoretical foundation and previously, seemingly antagonistic elements, such as Holland's Schema theorem and the Building Block hypothesis, have been reconciled with exact formulations, such as the Vose model [4]. In particular, coarse grained models have led to exact schema theorems, both in GA's [5] and GP [6], that generalize the traditional Schema theorem for GA's and show exactly when the Building Block hypothesis is valid.

This new unified formulation is, at heart, a coarse-grained one, where evolution of a string/program is written in terms of schemata/hyperschemata. However, it remains an exact description with no loss of precision. For GA's, this coarse-grained description was derived [5, 7] from the underlying microscopic, string-based model, thus showing that the Vose and coarse-grained models were just different representations of the former. It was later shown [8] that this change of formulation could be described as a coordinate transformation, passing from the basis of strings to that of building block schemata. In this paper we present a manifestly covariant form of the evolution equations for a canonical GA, showing how to pass between the three important bases: string, Walsh and building block, using linear coordinate transformations and describing the properties of the different operators - selection, mutation and recombination - in the different bases, paying particular attention to recombination in the BBB. We also show how the BBB is related to the Taylor basis.

2 Mathematical Preliminaries

We first consider some mathematical preliminaries. Readers unused to some of the notation familiar from linear algebra and tensor analysis may consult an appropriate introductory reference such as [9]. Consider a discrete, finite set A and the commutative algebra $\mathcal{F}_A$ of real-valued functions on A. We will refer to the elements of A as points. To each subset B of A there corresponds the *characteristic function* (CF) $f_B \in \mathcal{F}_A$, which takes the value 1 on each element of B and is zero elsewhere. Conversely, every function $f \in \mathcal{F}_A$, with values in $\{0, 1\}$, defines a subset of A as the locus of the points where it takes the value 1. Thus, one can denote subsets of A by listing the collection of points that make up the subset or by giving the corresponding characteristic function.

Given an n-dimensional vector space V, with basis $B_V = \{f^1, \ldots, f^n\}$. The set V^* of real-valued linear functionals on V is also an n-dimensional vector space, called the *dual* of V. The duality is via a bilinear *inner product*, $\langle \cdot, \cdot' \rangle : V^* \otimes V \to \mathbb{R}$, $x \otimes a \mapsto \langle x, a \rangle \equiv x(a)$. A basis $B_{V^*} = \{e_1, \ldots, e_n\}$ is called *dual* to B_V if $\langle e_i, f^j \rangle = \delta_i^j$. The element $C = e_i \otimes f^i \in V^* \otimes V$ is called the *canonical element* and satisfies

$$\langle C, a \otimes \mathrm{id} \rangle \equiv \langle e_i, a \rangle \, f^i = a \,, \qquad \langle \mathrm{id} \otimes x, C \rangle \equiv \langle x, f^i \rangle \, e_i = x \,, \tag{1}$$

for all a in V, x in V^*. Under a linear change of basis of V, associated with a coordinate transformation matrix Λ, $\mathbf{f}$ and its dual basis $\mathbf{e}$ transform as

$$\mathbf{f} \to \mathbf{f}' = \Lambda \mathbf{f} \qquad \mathbf{e} \to \mathbf{e}' = \mathbf{e} \Lambda^{-1} \,, \tag{2}$$

where $\mathbf{f}$ is the column vector $(f^1, \ldots, f^n)^T$ and $\mathbf{e}$ is the row vector $(e_1, \ldots, e_n)$.

3 Bases in the Space of Characteristic Functions

We now consider a number of bases in the configuration space of fixed-length, binary strings[1]. In this case the natural configuration space is an N-dimensional unit cube C_N, which can be embedded in $\mathbb{R}^N$, with coordinates $\{x_1, \ldots, x_N\}$, so that the points $(0, 0, \ldots, 0)$ and $(1, 1, \ldots, 1)$ are antipodal points of the cube. Then, restricted to C_N, each x_i, $1 \leq i \leq N$, takes the values 0 and 1. Define $\bar{x}_i \equiv e - x_i$, where e is the unit function on C_N, taking the value 1 on each vertex. In what follows, unless explicitly mentioned otherwise, all functions are considered restricted to C_N. In that case, one may impose algebraic relations on the coordinate functions, compatible with their allowed numerical values,

$$x_i^2 = x_i \,, \qquad \bar{x}_i^2 = \bar{x}_i \,, \qquad x_i \, \bar{x}_i = 0 \,. \tag{3}$$

[1] Generalizations to higher cardinality alphabets are rather straightforward, *e.g.*, for a 2^M-cardinality alphabet one may simply consider a block of M consecutive bits as a single "letter". Also, much of what is presented below should be generalizable to the case of variable-length strings and GP.

Notice that x_i is the CF for half of the cube (all vertices with $x_i = 1$) while $\bar{x}_i$ corresponds to the other half — then e corresponds to the entire cube, in accordance with $x_i + \bar{x}_i = e$. To specify lower dimensional k-cubes (hyperplanes), $k < N$, one needs products of the coordinates, *e.g.*, for $N = 3$, $x_1 x_2$ specifies the edge connecting the points $(1, 1, 0)$ and $(1, 1, 1)$, while $x_1 \bar{x}_2 \bar{x}_3$ specifies the point $(1, 0, 0)$. In general, there exists a one-to-one correspondence between monomials of degree $N - k$ in x_i, $\bar{x}_i$ and k-cubes.

3.1 The δ-basis

The standard basis in $\mathcal{F}_{C_N}$ is the *δ-basis* B_δ, consisting of the CF's of all 2^N vertices of C_N, *i.e.*, of delta-like functions with support on the vertices of the cube,

$$B_\delta = \{\bar{x}_1 \bar{x}_2 \dots \bar{x}_N,\ \bar{x}_1 \bar{x}_2 \dots x_N,\ \dots,\ x_1 x_2 \dots x_N\}\,. \tag{4}$$

We have singled out above the point $(1, 1, \dots, 1)$, its CF being the last element of the basis. The same construction can be based on an arbitrary vertex by defining

$$B_\delta^P = \{\bar{\alpha}_1 \bar{\alpha}_2 \dots \bar{\alpha}_N,\ \bar{\alpha}_1 \bar{\alpha}_2 \dots \alpha_N,\ \dots,\ \alpha_1 \alpha_2 \dots \alpha_N\}\,, \tag{5}$$

where the CF of the vertex P is $\alpha_1 \alpha_2 \dots \alpha_N$, with each of the α_i being either x_i or $\bar{x}_i$, and defining the bar operation to be involutive ($\bar{\bar{x}} = x$).

The particular ordering we choose above is "odometer"-like: referring to the choice $P = (1, 1, \dots, 1)$, we start at the origin (the antipode of P in the cube), with CF $\bar{x}_1 \bar{x}_2 \dots \bar{x}_N$, and let the last factor take on all possible values ($\bar{x}_N$ and x_N, in this case), then the next-to-last factor advances *etc.*. This is the standard ordering for tensor products of vector spaces: if $\{v_1, \dots, v_n\}$ and $\{w_1, \dots, w_m\}$ are bases for V and W respectively, the basis for the tensor product $V \otimes W$ is taken to be $\{v_1 \otimes w_1,\ v_1 \otimes w_2, \dots,\ v_1 \otimes w_m, v_2 \otimes w_1, \dots,\ v_n \otimes w_m\}$, where, in our case, a k-cube is considered as a k-fold tensor product of 1-cubes.

3.2 The monomial basis

We introduce here the *monomial basis* B_{m}^P, associated to the vertex P of the unit cube. It consists of the CF's of all k-cubes containing P — these are all monomials in the variables that appear in the CF of P. Anticipating the discussion of recombination in Sect. 5.1, we point out that the monomial basis is isomorphic to the BBB which, as was shown in [8], most naturally enters in the description of recombination. We clarify the construction of B_{m}^P with a couple of examples.

Example 1 *The δ and monomial bases for $N = 1$ and $N = 3$*

For $N = 1$, the δ-basis of the previous subsection (corresponding to the vertex with CF x_1) is $B_\delta^{x_1} = \{\bar{x}_1,\ x_1\}$, while the monomial basis (corresponding to the same vertex) is $B_{\mathrm{m}}^{x_1} = \{e,\ x_1\}$. Arranging the basis elements in columns, $\mathbf{x}_\delta^{x_1} = (\bar{x}_1, x_1)^T$, $\mathbf{x}_{\mathrm{m}}^{x_1} = (e, x_1)^T$ we have,

$$\mathbf{x}_{\mathrm{m}}^{x_1} = \Lambda \mathbf{x}_\delta^{x_1}\,, \tag{6}$$

where $\Lambda \equiv \begin{pmatrix} 1 & 1 \\ 0 & 1 \end{pmatrix}$ does not depend on the chosen vertex P.

For $N = 3$, the vertex $(1, 0, 0)$, with CF $x_1\bar{x}_2\bar{x}_3$, induces the basis

$$B_{\mathrm{m}}^{x_1\bar{x}_2\bar{x}_3} = \{e,\, \bar{x}_3,\, \bar{x}_2,\, \bar{x}_2\bar{x}_3,\, x_1,\, x_1\bar{x}_3,\, x_1\bar{x}_2,\, x_1\bar{x}_2\bar{x}_3\}\,, \tag{7}$$

consisting of the CF's of all k-cubes, with $0 \leq k \leq N = 3$, containing the above vertex, ranging from the entire 3-cube to the vertex itself. The use of the standard tensor product ordering for the basis elements, mentioned earlier, becomes clear if one substitutes e's for the missing coordinates in each of the above monomials, *i.e.*, writing the basis as $\{eee, ee\bar{x}_3,\, e\bar{x}_2e, \ldots\}$. The matrix Λ_3 effecting the transition between the two bases is clearly the tensor cube of Λ, $\Lambda_3 = \Lambda^{\otimes 3} \equiv \Lambda \otimes \Lambda \otimes \Lambda$, where, in general, $(A \otimes B)_{ij,kl} = A_{ik}B_{jl}$, giving in the special case of 2×2 matrices

$$A \otimes B = \begin{pmatrix} aB & bB \\ cB & dB \end{pmatrix} = \begin{pmatrix} ax & ay & bx & by \\ az & aw & bz & bw \\ cx & cy & dx & dy \\ cz & cw & dz & dw \end{pmatrix}\,, \quad A \equiv \begin{pmatrix} a & b \\ c & d \end{pmatrix}\,, \quad B \equiv \begin{pmatrix} x & y \\ z & w \end{pmatrix}\,. \tag{8}$$

□

In the general case, a vertex P with CF $\alpha_1\alpha_2\cdots\alpha_N$, induces the monomial basis

$$B_{\mathrm{m}}^{\alpha_1\cdots\alpha_N} = \{e,\, \alpha_N,\, \ldots,\, \alpha_1\ldots\alpha_{N-2}\alpha_N,\, \alpha_1\ldots\alpha_{N-1},\, \alpha_1\ldots\alpha_{N-1}\alpha_N\}\,, \tag{9}$$

with

$$\mathbf{x}_{\mathrm{m}}^P = \Lambda_N \mathbf{x}_\delta^P\,, \qquad\qquad \Lambda_N \equiv \Lambda^{\otimes N}\,. \tag{10}$$

Λ_N does not depend on P as long as the two bases are chosen according to Eqs. (5) and (9). From Eq. (10) we have

$$\Lambda_{n+1} = \Lambda \otimes \Lambda_n = \begin{pmatrix} \Lambda_n & \Lambda_n \\ 0 & \Lambda_n \end{pmatrix}\,. \tag{11}$$

The matrix elements, Λ_{IJ}, are such that $\Lambda_{IJ} = 1$ if the vertex J is contained in the k-cube I, and is zero otherwise.

3.3 The Walsh basis

The other basis that has been extensively studied is the *Walsh basis*. As in the previous section, the N-bit case can be obtained by tensoring up the 1-bit one, the latter being illustrated by the following example.

Example 2 *The δ and Walsh bases for $N = 1$*

For $N = 1$, the δ-basis of subsection 3.1 (corresponding to the vertex with CF x_1) is $B_\delta^{x_1} = \{\bar{x}_1,\, x_1\}$, while the corresponding Walsh basis is $B_{\mathrm{W}}^{x_1} = \{(\bar{x}_1 + x_1)/\sqrt{2},\, (\bar{x}_1 - x_1)/\sqrt{2}\}$. Thus we have,

$$\mathbf{x}_{\mathrm{W}}^{x_1} = \Lambda_W \mathbf{x}_\delta^{x_1}\,, \tag{12}$$

where $\Lambda_W \equiv 2^{-1/2}\begin{pmatrix} 1 & 1 \\ 1 & -1 \end{pmatrix}$ does not depend on P. □

4 The Dual Description: Bases in the Space of Points

We turn now from the space $\mathrm{Fun}(C_N)$, of real valued functions on C_N, to its dual (as a vector space), whose elements are naturally identified with (linear combinations of) the vertices of the cube, viewed as geometrical points in $\mathbb{R}^N$. The duality is via pointwise evaluation, *i.e.*, given a function f and a vertex g, their inner product is simply the value of f at g, $\langle g, f\rangle = f(g)$, extended by linearity in each of the arguments. We now consider the duals of the δ and the monomial bases of $\mathrm{Fun}(C^N)$ — we do not include the Walsh basis since $\Lambda_W^{-1} = \Lambda_W$.

4.1 The vertex basis

We define the *vertex basis* B_{v}, as the dual of B_δ — it clearly consists of the vertices g_R of the cube, appropriately ordered, $B_{\mathrm{v}} \equiv B_\delta^* = \{g_R\}$. R here is a composite index, $R = (r_1 \dots r_n)$, with each r_i being either 0 or 1. Taking for concreteness the reference point $P = (11\dots1)$, we arrange the vertices in a row vector, $\mathbf{g}_{\mathrm{v}} = (g_{00\dots0}, g_{00\dots01}, \dots, g_{11\dots1})$.

4.2 The Taylor basis

Dual to the basis B_{m} of the k-cube CF's (arranged in the column vector $\mathbf{x}_{\mathrm{m}} = \Lambda_N \mathbf{x}_\delta$) is the *Taylor basis* $B_{\mathrm{T}} \equiv B_{\mathrm{m}}^*$, given in (row) vector form by $\mathbf{g}_{\mathrm{T}} = \mathbf{g}_{\mathrm{v}} \Lambda_N^{-1}$. To illustrate its geometrical meaning consider the following[2]

Example 3 *δ, monomial, vertex and Taylor bases for N=2*

We have

$$\mathbf{x}_\delta = \begin{pmatrix} \bar{x}_1\bar{x}_2 \\ \bar{x}_1 x_2 \\ x_1\bar{x}_2 \\ x_1 x_2 \end{pmatrix}, \qquad \mathbf{x}_{\mathrm{m}} = \Lambda_2 \mathbf{x}_\delta = \begin{pmatrix} 1&1&1&1 \\ 0&1&0&1 \\ 0&0&1&1 \\ 0&0&0&1 \end{pmatrix} \begin{pmatrix} \bar{x}_1\bar{x}_2 \\ \bar{x}_1 x_2 \\ x_1\bar{x}_2 \\ x_1 x_2 \end{pmatrix} = \begin{pmatrix} e \\ x_2 \\ x_1 \\ x_1 x_2 \end{pmatrix}, \quad (13)$$

while, for the dual bases, we compute

$$\begin{aligned} \mathbf{g}_{\mathrm{v}} &= \big(g_{00}, g_{01}, g_{10}, g_{11}\big) \\ \mathbf{g}_{\mathrm{T}} = \mathbf{g}_{\mathrm{v}} \Lambda_2^{-1} &= \big(g_{00}, g_{01}, g_{10}, g_{11}\big) \begin{pmatrix} 1&-1&-1&1 \\ 0&1&0&-1 \\ 0&0&1&-1 \\ 0&0&0&1 \end{pmatrix} \\ &= \big(g_{00},\ g_{01} - g_{00},\ g_{10} - g_{00},\ g_{11} - g_{10} - g_{01} + g_{00}\big)\ . \end{aligned} \quad (14)$$

[2] The Taylor basis was considered, in a different context, by Weinberger in [10].

Apply now the first of (1) to an arbitrary function $f \in \mathrm{Fun}(C_N)$, using the pair of dual bases B_m, B_T,

$$\begin{aligned} f &= \langle (\mathbf{g}_\mathrm{T})_i, f\rangle \, (\mathbf{x}_\mathrm{m})^i \\ &= f(g_{00})\, e + \big[f(g_{01}) - f(g_{00})\big]\, x_2 + \big[f(g_{10}) - f(g_{00})\big]\, x_1 \\ &\quad + \big[f(g_{11}) - f(g_{10}) - f(g_{01}) + f(g_{00})\big]\, x_1 x_2 \,. \end{aligned} \tag{15}$$

Thus, one obtains the Taylor expansion of f around the origin, with

$$\partial_1 \equiv g_{10} - g_{00}\,, \qquad \partial_2 \equiv g_{01} - g_{00}\,, \qquad \partial_{12} \equiv g_{11} - g_{10} - g_{01} + g_{00}\,, \tag{16}$$

being discrete derivative operators which, given that f is at most linear in the x_i's, coincide with the exact ones. □

Since, for higher N, B_m still consists of all monomials in the x_i's, it is easy to see that the above interpretation of the elements of B_T persists for all N.

5 Genetic dynamics in the different coordinate systems

We now consider dynamical evolution in the different bases in the presence of the genetic operators of selection, mutation and recombination, focusing primarily, due to lack of space, on the latter. The state of the system is described by the 2^N-component vector $\mathbf{P}$, the physical interpretation of which is basis-dependent. The equation governing its time evolution is

$$P_I(t+1) = \sum_J M_I{}^J P^c_J(t) \tag{17}$$

where $(M_I{}^J)$ is the mutation matrix and $P^c_I(t)$ is given by[3]

$$P^c_I = (1-p_c)P'_I + p_c(P'_I + G_I - L_I)\,, \tag{18}$$

p_c being the probability that recombination takes place. In the δ-basis, P^c_I is the probability to find the string I after selection and recombination and P'_I is the probability to select I. The gains term G_I counts the total number of children of type I produced while disappearing parents of type I are counted by the losses term L_I — this organization of terms is different from the one used in past work [5, 7] but we find that it facilitates the counting. Indeed, L_I is clearly equal to P'_I, since, by our definition, every parent participating in recombination is lost, while G_I can be written as

$$G_I = \sum_M \sum_{J,K} \frac{1}{2}\big(p(M) + p(\bar{M})\big)\lambda_I{}^{JK}(M) P'_J P'_K\,, \tag{19}$$

where $\lambda_I{}^{JK}(M)$ is an interaction term between objects I, J and K, M is a recombination mask, occuring with probability $p(M)$, and $\bar{M}$ denotes the conjugate

[3] Henceforth all time-dependent quantities are evaluated at time t, unless explicitly shown otherwise.

mask, *i.e.*, such that $M + \bar{M} = (11\ldots1)$ — we explain further the form of (19) in Sect. 5.1 below. Substituting the expressions for G_I and L_I, (17) becomes

$$P_I(t+1) = {M_I}^J \left((1-p_c)P'_J + p_c \sum_M \sum_{K,L} \frac{1}{2}\big(p(M) + p(\bar{M})\big) {\lambda_J}^{KL}(M) P'_K P'_L \right) . \tag{20}$$

Despite the covariance of (20), the facility of its analysis as well as its physical interpretation are basis-dependent. The dynamics is governed by the mutation matrix ${M_I}^J$, the tensor ${\lambda_I}^{JK}(M)$, the mask probability distribution $p(M)$ and the fitness values f_I, hidden inside P'_I. In this sense the evolutionary algorithm is a "black box" whose output depends on a large set of parameters. It therefore behooves us to look for symmetries and regularities that may be exploited to effect a natural coarse graining, making manifest the effective degrees of freedom of the dynamics.

5.1 Recombination

We will now consider recombination in the δ- and BB bases. For a discussion of recombination in the Walsh basis, in a much different context, see [11].

Recombination in the δ ("string") basis. In this basis, P_I is the probability (relative population) of the string I. For each mask M, there are generally several pairs of parent strings $\{J, K\}$ that produce I as their child. The tensor $\lambda(M)$ in Eq. (19) is given by

$${\lambda_I}^{JK}(M) = \prod_{r=1}^{N} [1 + i_r + j_r + m_r(j_r + k_r)] \mod 2 \tag{21}$$

which is 1 if the first child of the recombination of J, K, with mask M, is I, and zero otherwise (we use the convention that a 0 in the mask denotes that the first child obtains the corresponding bit from the first parent). Then ${\lambda_I}^{KJ}(M) = {\lambda_I}^{JK}(\bar{M})$ checks whether I is being produced as a second child. One may define a mask-independent average λ_I by ${\lambda_I}^{JK} = \sum_M p(M) {\lambda_I}^{JK}(M)$ whereupon (19) becomes, in matrix notation,

$$G_I(t) = \mathbf{P}^T R_I \mathbf{P} , \qquad R_I \equiv \frac{1}{2}(\lambda_I + \lambda_I^T) . \tag{22}$$

Notice that the second of (22) is valid in all bases, since both matrix indices of λ_I are contravariant (upper). For reasons explained in Sect. 5.2, λ_I is a more convenient object to work with than R_I. Again, the covariance of (22) guarantees its validity in all bases, with ${R_I}^{JK}$ transforming, along with ${\lambda_I}^{JK}$, as a rank-three tensor (see Ex. 5 below). Ignoring selection and mutation, Eq. (20) then becomes

$$P_I(t+1) = (1-p_c)P_I + p_c \mathbf{P}^T R_I \mathbf{P} . \tag{23}$$

Example 4 *$N = 2$ recombination in the δ-basis*

We fix $I = (11)$ and take $p(M) = 1/4$ (independent of M). From (21) we compute

$$\lambda_{(11)}\big((00)\big) = \begin{pmatrix} 0&0&0&0\\ 0&0&0&0\\ 0&0&0&0\\ 1&1&1&1 \end{pmatrix}, \qquad \lambda_{(11)}\big((01)\big) = \begin{pmatrix} 0&0&0&0\\ 0&0&0&0\\ 0&1&0&1\\ 0&1&0&1 \end{pmatrix}, \tag{24}$$

while $\lambda_{(11)}\big((10)\big) = \lambda_{(11)}\big((01)\big)^T$ and $\lambda_{(11)}\big((11)\big) = \lambda_{(11)}\big((00)\big)^T$. Then

$$\lambda^{\delta}_{(11)} = \frac{1}{4}\begin{pmatrix} 0&0&0&1\\ 0&0&1&2\\ 0&1&0&2\\ 1&2&2&4 \end{pmatrix}, \tag{25}$$

where we have reinstated a (so far suppressed) superscript δ to remind us of the basis, and $R^{\delta}_{(11)} = \lambda^{\delta}_{(11)}$. Eq. (23) then gives

$$P_{(11)}(t+1) = P_{(11)} + \frac{p_c}{2}\left(P_{(10)}P_{(01)} - P_{(11)}P_{(00)}\right). \tag{26}$$

The equations for the other strings are obtained by renaming the indices. □

Recombination in the monomial ("building block") basis. As the above example shows, recombination is rather complicated in the δ-basis. A more efficient organization of the various terms that contribute to $G_I(t)$ can be achieved if one thinks in terms of Building Block *schemata*. For example, (11) can be obtained by recombining the schemata $(1*)$ and $(*1)$, where a $*$ denotes *any* bit. Each string gives rise to 2^N schemata associated with it, by all possible substitutions of its bits by $*$'s — the corresponding set of schemata constitutes the BBB for that string. For example, (11) generates the basis $\{(**), (*1), (1*), (11)\}$. Recombination involves the interaction of conjugate schemata only[4], so one expects some sort of "skew diagonalization" of the process in this basis. To connect with the discussion in Sect. 3.2, notice that substitution of a particular bit by a $*$ corresponds, at the level of CF's, to substitution of a coordinate x_i (or $\bar{x}_i$) by the unit function e. It is then clear that the CF's of the Building Blocks are exactly the elements of the monomial basis of Sect. 3.2. We conclude that *the Taylor basis is dual to the BBB.*

The CF corresponding to a schema is the sum of the CF's of all vertices (strings) that the schema matches. On the other hand, it is clear that the probability of a certain schema is likewise the sum of the probabilities of all strings

[4] We define the conjugation $\bar{\ }$ of schemata: the string $R = (r_1 r_2 \dots r_N)$ generates the basis $\{(**\dots*), (**\dots r_N), \dots, (r_1 r_2 \dots r_N)\}$ and $\bar{r}_i = *$ while $\bar{*} = r_i$, if the $*$ is in position i — the conjugate of a schema is the schema with conjugate bits.

that the schema matches. This implies that, in going from one basis to another, probabilities transform like CF's — in particular

$$\mathbf{P}^{\mathrm{m}} = \Lambda_N \mathbf{P}^{\delta} . \tag{27}$$

Example 5 *$N = 2$ recombination in the monomial basis*

One can calculate the mask-averaged interaction term in the BBB, $(\lambda^{\mathrm{m}})_I{}^{JK}$, by transforming λ^{δ} as a rank-three tensor,

$$(\lambda^{\mathrm{m}})_I{}^{JK} = (\lambda^{\delta})_{I'}{}^{J'K'} (\Lambda_2)_I{}^{I'} (\Lambda_2^{-1})_{J'}{}^{J} (\Lambda_2^{-1})_{K'}{}^{K} , \tag{28}$$

to find, for example, for $\lambda^{\mathrm{m}}_{(11)}$

$$\lambda^{\mathrm{m}}_{(11)} = \frac{1}{4} \begin{pmatrix} 0\,0\,0\,1 \\ 0\,0\,1\,0 \\ 0\,1\,0\,0 \\ 1\,0\,0\,0 \end{pmatrix} . \tag{29}$$

As expected, it is skew diagonal. The dynamical equation for $P_{(11)}(t)$ is

$$P_{(11)}(t+1) = (1 - \frac{p_c}{2}) P_{(11)} + \frac{p_c}{2} P_{(*1)} P_{(1*)} , \tag{30}$$

which by substituting $P_{(*1)} = P_{(11)} + P_{(01)}$, and analogously for $P_{(1*)}$, can be seen to coincide with (26). □

The above result generalizes to arbitrary N (see Sect. 5.2 below)

$$(\lambda^{\mathrm{m}})_{(11\ldots1)}{}^{JK} = 2^{-N} \delta^{J,\, 2^N+1-K} . \tag{31}$$

In the δ-basis, the equations for the other elements of the basis can be obtained from the one for $(11\ldots1)$ by renaming the indices. In the monomial basis, the situation is even simpler: one obtains, for example, the equation for $(11*)$ from the one for (11) simply by attaching an extra $*$ to all indices - this generalizes in the obvious way to any number of $*$'s in any position, so that (31), inserted in (23), gives essentially the equations for all basis elements, for all N.

5.2 The tensor product structure of recombination

As we have seen above, the dynamics of recombination is controlled by the tensor $\lambda(M)$, which contains the information about which parents may give rise to a particular child. In deciding this, one needs to perform a bit-by-bit test, the outcome for the entire string being the logical AND of the individual bit tests (see Eq. 21, where AND corresponds to multiplication). The fact that the value of $\lambda(M)$ factorizes in this manner, reflects itself in that $\lambda_I{}^{JK}(M)$, for a length-N string, is the tensor product of the λ's of its bits,

$$\lambda_I(M) = \prod_{r=1}^{N} \lambda_{(i_r)}{}^{(j_r)(k_r)}((m_r)) , \tag{32}$$

or, in matrix notation,

$$\lambda_I{}^{JK}(M) = \lambda_{(i_1)}((m_1)) \otimes \lambda_{(i_2)}((m_2)) \otimes \ldots \otimes \lambda_{(i_N)}((m_N))\,. \tag{33}$$

A simple calculation then shows that the same is true for the mask-independent λ, *i.e.*, in matrix notation, $\lambda_I = \lambda_{(i_1)} \otimes \ldots \otimes \lambda_{(i_N)}$. Finally, given that Λ_N is itself the N-th tensor power of the 1-bit Λ_1, we conclude that the above statements about λ are valid in all bases. Notice that $R_I{}^{JK}$ *does not* factorize in this manner — this is because checking for the first *or* the second child, for $N > 1$, is not a bit-wise operation.

Example 6 *The tensor product structure in the string and Building Block bases*

Consider $N = 1$ recombination in the string basis. We find

$$\lambda^{\delta}_{(0)} = \frac{1}{2}\begin{pmatrix} 2 & 1 \\ 1 & 0 \end{pmatrix}\,, \qquad \lambda^{\delta}_{(1)} = \frac{1}{2}\begin{pmatrix} 0 & 1 \\ 1 & 2 \end{pmatrix}\,. \tag{34}$$

Transforming to the BBB we find

$$\lambda^{\mathrm{m}}_{(*)} = \begin{pmatrix} 1 & 0 \\ 0 & 0 \end{pmatrix}\,, \qquad \lambda^{\mathrm{m}}_{(1)} = \frac{1}{2}\begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix}\,. \tag{35}$$

The second equation above clearly shows that $\lambda^{\mathrm{m}}_{(11\ldots1)} = (\lambda^{\mathrm{m}}_{(1)})^{\otimes N}$ is skew-diagonal for all N. Notice also that the $\lambda^{\delta}_{(11)}$ given in Eq. (25) is just the tensor square of $\lambda^{\delta}_{(1)}$ given in Eq. (34) above. □

Much of our discussion so far referred to the case of equally probable masks. The above results however are also valid for the case of uniform crossover, where the first child gets the i-th bit from the first parent with probability p_i, resulting in the mask probability distribution $p(M) = \prod_{i\in I_0} p_i \prod_{j\in I_1}(1-p_j)$, where I_α is the subset of indices in I with value α. For example, for $N = 2$, we get

$$p_{(00)} = p_1p_2\,, \quad p_{(01)} = p_1(1-p_2)\,, \quad p_{(10)} = (1-p_1)p_2\,, \quad p_{(11)} = (1-p_1)(1-p_2)\,.$$

Then the average λ still factorizes, with the string basis 1-bit factors

$$\lambda^{\delta}_{(0)} = \begin{pmatrix} 1 & p_i \\ 1-p_i & 0 \end{pmatrix}\,, \qquad \lambda^{\delta}_{(1)} = \begin{pmatrix} 0 & 1-p_i \\ p_i & 1 \end{pmatrix}\,, \tag{36}$$

and their BBB counterparts

$$\lambda^{\mathrm{m}}_{(*)} = \begin{pmatrix} 1 & 0 \\ 0 & 0 \end{pmatrix}\,, \qquad \lambda^{\mathrm{m}}_{(1)} = \begin{pmatrix} 0 & 1-p_i \\ p_i & 0 \end{pmatrix}\,, \tag{37}$$

where i denotes the position of the bit. We see again that $\lambda^{\mathrm{m}}_{(11\ldots1)}$ is skew-diagonal. It is easy to see that this generalizes to *any* probability distribution $p(M)$ since $\lambda^{\mathrm{m}}_{(11\ldots1)}(M)$ itself is skew-diagonal (a fact that does not depend on $p(M)$) and $\lambda_I{}^{JK}$ is a sum over such matrices. It is for this reason that BBB makes manifest the effective degrees of freedom for recombination — the Building Block schemata themselves.

5.3 Selection and mutation

In Eq. (20), selection is "hidden" inside $P'_I(t)$. In the absence of any further information, all that can be said is that $\mathbf{P}'$ transforms like a vector. When $\mathbf{P}'$ is given in terms of a fitness matrix, $P'_I = \sum_J F_I{}^J P_J$, we may infer that, under a change of basis, $F \to \Lambda F \Lambda^{-1}$. In the δ-basis, F is taken to be diagonal. In the Walsh basis, F is complicated, the number of non-zero elements depending on the degree of epistasis in the landscape. In the BBB, $F^{\mathrm{m}} = \Lambda F^{\delta} \Lambda^{-1}$ is not diagonal, however, it can be shown that $F^{\mathrm{m}} = F^{\mathrm{m}\prime} + A$, where $F^{\mathrm{m}\prime}$ is diagonal and $A\mathbf{P}^m = 0$, hence the dynamics is given essentially by a diagonal matrix, as in the δ-basis.

For proportional selection, $(F^{\mathrm{m}\prime})_I{}^J = (f_I(t)/\bar{f}(t))\delta_I{}^J$, where $f_I(t)$ is the fitness of the Building Block I and is population- (and hence time-) dependent. Interestingly enough, $A\mathbf{P}^m = 0$, hence the dynamics is given essentially by a diagonal matrix, as in the δ-basis. However, the algebraic relation between the two sets of diagonal elements is non-trivial. Note also that only in the very restrictive case of a multiplicative fitness landscape can one generate the N-bit problem from the tensor product of N 1-bit problems.

The mutation matrix transforms like $M \to \Lambda M \Lambda^{-1}$. When the mutation probability p_i of the i-th bit is independent of the other bits, the N-bit mutation matrix factorizes in 1-bit factors,

$$M_N = M(p_1) \otimes M(p_2) \otimes \ldots \otimes M(p_N)\,, \tag{38}$$

where $M(p_i) \equiv \begin{pmatrix} (1-p_i) & p_i \\ p_i & (1-p_i) \end{pmatrix}$. The factorizability of M_N is then preserved in all bases,

$$M_N \to \Lambda_N M_N {\Lambda_N}^{-1} = \Lambda_1 M(p_1) {\Lambda_1}^{-1} \otimes \ldots \otimes \Lambda_1 M(p_N) {\Lambda_1}^{-1}\,. \tag{39}$$

In the Walsh basis, $M_1^W = \begin{pmatrix} 1 & 0 \\ 0 & 1-2p_i \end{pmatrix}$, while in the BBB, $M_1^m = \begin{pmatrix} 1 & 0 \\ p_i & 1-2p_i \end{pmatrix}$. Thus, as is well known, in the Walsh basis the N-bit mutation matrix is diagonal, while in the BBB it is triangular. In both cases the eigenvalues can simply be read off from the diagonal.

6 Conclusions

We presented GA dynamics in a covariant form, showing how different existing formulations — string, Walsh mode, Building Block schemata — can be related by linear coordinate transformations. It was shown that the N-bit transformation matrices are the N-th tensor power of the corresponding 1-bit matrix. The manifest covariance of the dynamical equations guarantees their validity in all bases — nevertheless, the analysis and its interpretation can be greatly simplified by choosing the basis best adapted to the genetic operator under study. The string basis is convenient for selection-dominated dynamics, while that of Walsh

is natural for dynamics dominated by mutation. In this paper we concentrated on the most complicated operator — recombination — showing how the BBB offered the most natural description, the effective degrees of freedom of recombinative dynamics being Building Block schemata. Introducing a description in terms of characteristic functions in configuration space, we showed that the BBB is dual to the standard Taylor basis — the presented mathematical framework, we believe, clarifies several conceptually obscure points. A thorough analysis of the factorizability of the various operators was given, resulting in an enormous simplification of their calculation in the different bases. With the unification program for EC in mind, straightforward generalizations to the case of higher cardinality alphabets and variable-length strings have been alluded to. Given the great similarity between the coarse-grained formulations of GA's and GP, it is reasonable to expect that the above coordinate transformations have analogues in the GP case.

Acknowledgements

The authors acknowledge support from DGAPA-PAPIIT projects IN114302 (CC), ES100201 (CRS) and CONACyT projects 41208-F (CC), 30422-E (CRS).

References

1. W. B. Langdon and R. Poli. *Foundations of Genetic Programming*. Springer Verlag, Berlin, New York, 2002.
2. C. R. Stephens and R. Poli. E C theory - in theory: Towards a unification of evolutionary computation theory. In A. Menon, editor, *Frontiers of Evolutionary Computation*, pages 129–156. Kluwer Academic Publishers, 2004.
3. C. R. Stephens and A. Zamora. EC theory: A unified viewpoint. In Erick Cantú Paz, editor, *Proceedings of GECCO 2003*, pages 1394–1402, Berlin, Germany, 2003. Springer Verlag.
4. Michael D. Vose. *The simple genetic algorithm: Foundations and theory*. MIT Press, Cambridge, MA, 1999.
5. C. R. Stephens and H. Waelbroeck. Schemata evolution and building blocks. *Evol. Comp.*, 7:109–124, 1999.
6. Riccardo Poli. Exact schema theory for genetic programming and variable-length genetic algorithms with one-point crossover. *Genetic Programming and Evolvable Machines*, 2(2):123–163, 2001.
7. C. R. Stephens and H. Waelbroeck. Effective degrees of freedom of genetic algorithms and the block hypothesis. In T. Bäck, editor, *Proceedings of ICGA97*, pages 31–41, San Francisco, CA, 1997. Morgan Kaufmann.
8. Christopher R. Stephens. The renormalization group and the dynamics of genetic systems. *Acta Phys. Slov.*, 52:515–524, 2003.
9. M.A. Akivis and V.V. Goldberg. *An Introduction to Linear Algebra and Tensors*. Dover Publications, Mineola, NY, 1977.
10. Edward D. Weinberger. Fourier and Taylor series on fitness landscapes. *Biological Cybernetics*, 65:321–330, 1991.
11. Alden H. Wright. The exact schema theorem. http://www.cs.umt.edu/CS/FAC-/WRIGHT/papers/schema.pdf, January 2000.