# Modeling Variation in Cooperative Coevolution Using Evolutionary Game Theory

**R. Paul Wiegand**
Department of Computer Science
George Mason University
Fairfax, VA 22030
paul@tesseract.org

**William C. Liles**
Department of Computer Science
George Mason University
Fairfax, VA 22030
wliles@gmu.edu

**Kenneth A. De Jong**
Department of Computer Science
George Mason University
Fairfax, VA 22030
kdejong@gmu.edu

## Abstract

Though coevolutionary algorithms are currently used for optimization purposes, practitioners are often plagued with difficulties due to the fact that such systems frequently behave in counter intuitive ways that are not well understood. This paper seeks to extend work which uses evolutionary game theory (EGT) as a form of dynamical systems modeling of coevolutionary algorithms in order to begin to answer questions regarding how these systems work. It does this by concentrating on a particular subclass of cooperative coevolutionary algorithms, for which multi-population symmetric evolutionary game theoretic models are known to apply. We examine dynamical behaviors of this model in the context of static function optimization, by both formal analysis, as well as model validation study. Finally, we begin looking at the effects of variation by extending traditional EGT, offering some introductory analysis, as well as model validation. In the course of this study, we investigate the effects of parameterized uniform crossover and bit-flip mutation.

# 1 INTRODUCTION

The task of understanding coevolutionary algorithms (CEAs) is very difficult due to the dynamic nature of their fitness landscapes. One of the more promising recent developments has been the use of evolutionary game theory (EGT) to provide a dynamical systems analysis of CEAs (Ficici and Pollack, 2000b). However, similar to Markov chain analyses of standard EAs, EGT-based theory of CEAs must deal effectively with several issues if it hopes to provide strong, practical insights.

The first and most obvious issue is that most basic forms of EGT models make many simplifying assumptions in order to make analysis more tractable. These include assumptions of infinite population sizes, complete pair-wise interactions among individuals in the population(s), and a lack of variational operators. With such strong assumptions EGT models improve our understanding of simple CEA dynamics, but don't shed much light on the kinds of CEAs practitioners use. In addition, practitioners are likely to be using CEAs to solve computational problems, and EGT models only indirectly address questions such as how effective are CEAs at solving difficult optimization problems.

This paper addresses both of these issues by relaxing one of the most restrictive EGT assumptions (no reproductive variation) and by relating CEA dynamics to issues of convergence to local/global optima. We do so by focusing on *cooperative* CEAs (CCEAs) (Potter and De Jong, 2000) which appear to us to be somewhat more tractable than competitive CEAs due (in part) to the existence of a clear and obvious objective measure (which is not generally true in competitive models). We build on recent work that has shown that a particular class of EGT models, *multi-population symmetric games (MPS)*, can be used to analyze the dynamical behaviors of CCEAs (Wiegand et al., 2002). This paper extends that research by making use of existing EA dynamical systems theory tools which model variational operators by using a type of *mixing function* (Vose, 1999). We then perform an initial formal analysis of MPS systems both with and without variation.

We begin by laying out some background with respect to how and what CCEA we are modeling, as well as a short discussion of our sample problem domain. Then we provide some analysis of the system without variation. In the fourth section, we describe how the model can be extended to include variation, and begin our analysis of variation by discussing the effects of uniform crossover and bit-flip mutation, then the combination of the two operators. Finally, we conclude by discussing what future steps should be taken in order to bring the full utility of the EGT analysis tool to bear on CCEA models which include variation in general, and crossover in particular.

# 2 MODELING COOPERATIVE COEVOLUTION

Throughout this paper the following notional conventions will be used. Lower case letters will be used to represent real numbers and functions. Vectors will be lower case letters denoted as $\vec{x}$. Capital letters will be used to denote sets and matrices. Euclidean spaces will be denoted $\Re^n$, where $n$ is a positive integer and indicates the dimensionality of the space. Given a set of connected points in Euclidean space, $X$, the *interior* of is denoted as $int(X)$. The boundary of $X$ is denoted $bnd(X)$. The $n$ dimensional unit simplex is denoted $\Delta^n$. The cartesian product of two simplexes is denoted $\Delta^n \times \Delta^m$.

We will assume a binary representation is being used when we incorporate variation. More formally, $x_i$ refers to the $i^{th}$ element of the population vector, where $i$ is a binary string in the set of all strings of length $l$. That is, $i \in \Omega$, where $\Omega = \{0, 1\}^l$ and $\|\Omega\| = n$, where $n$ is the number of distinct genotypes represented in the population. The symbols $\oplus$ and $\otimes$ will represent *bit-wise exclusive or* and *bit-wise and* operations on binary strings, respectively.

### 2.1 MULTI-POPULATION EGT MODELS

In this paper we will continue to assume infinite populations, allowing us to leverage existing EA theory (Vose, 1999) and EGT theory (Weibull, 1992; Hofbauer and Sigmund, 1998; Maynard-Smith, 1982) in which populations consist of an infinite number of individuals, but a finite number of $n$ distinct possible genotypes. This allows one to represent the state of a population at any given time by a vector in the unit simplex and changes in a population over time as dynamical system trajectories in the simplex.

While single population models are common in EGT, we will be focusing entirely on multi-population models, in which individuals interact only with members of the other population, not their own. To keep things simple in this paper we show how a two population model is constructed. In such a model, a common way of expressing the payoff rewards from individual interactions is by using a pair of *payoff matrices*. In general, when individuals from the first population interact with individuals from the second, one payoff matrix is used (e.g. $A$), while the second population receives rewards defined by a second payoff matrix (e.g. $B$). The defining characteristic of MPS games is that they are symmetric with respect to their payoff matrices. For two population games, this means they are symmetric when $A = B^T$. The symmetry assumption allow for some subtle simplification of the mathematics involved. For example, simple algebraic expansion will show that the weighted average payoff of the first population is the same as that of the second, $\vec{x} \cdot A\vec{y} = \vec{y} \cdot A^T\vec{x}$.

Given two populations, $(\vec{x}, \vec{y}) \in \Delta^n \times \Delta^m$, representing ratios of genotypes in two infinite populations, the following equations are used to define the dynamical system for a cooperative coevolutionary algorithm (without variation) operating on these populations:

$$\vec{u} = A\vec{y} \tag{1}$$

$$\vec{w} = A^T\vec{x} \tag{2}$$

$$x_i' = \left(\frac{u_i}{\vec{x} \cdot A\vec{y}}\right) x_i \tag{3}$$

$$y_i' = \left(\frac{w_i}{\vec{x} \cdot A\vec{y}}\right) y_i \tag{4}$$

where $\vec{x}\,'$ and $\vec{y}\,'$ represent the new population distributions for the next generation. $A$ and $A^T$ describe the payoffs associated with each pair of possible interactions. We also write the same system in a more abstract way using the following two equations: $\mathcal{G}_x = \mathcal{S}\left(\mathcal{F}_x, \vec{x}\right)$ and $\mathcal{G}_y = \mathcal{S}\left(\mathcal{F}_y, \vec{y}\right)$. In this model, $\mathcal{S}$ represents fitness proportional selection, and $\mathcal{F}$ is a function which assigns some vector of relative fitnesses to the strategies. In the model shown above, given the payoff matrix $A$, $\vec{u} = \mathcal{F}_x(\vec{x}, \vec{y})$ and $\vec{w} = \mathcal{F}_y(\vec{x}, \vec{y})$

### 2.2 MODELING CCEAS WITH MPS GAMES

In this paper we focus on one way in which CCEAs have been used successfully to perform function optimization. In this approach, each population is assigned a specific argument of the function to represent, and individuals in a given population must collaborate with individuals from other populations in order to obtain a fitness value, the value of the objective function (Potter and De Jong, 1994).

A simple two argument function serves as an example. Suppose we would like to optimize the function $f(x, y) = x^2 + y^2$ using cooperative coevolution. We represent potential $x$ argument values in one population, and potential $y$ argument values in a second population. We would evolve the two populations separately (i.e., they do not interbreed), but when it comes time to evaluate an individual in the $x$ population, we will need to select collaborating individuals from the $y$ population in order to obtain a value from the objective function. The same process is true in reverse for the $y$ population, with respect to collaborations from $x$.

As recent research suggests, an MPS game is a direct analogy of such a CCEA (Wiegand et al., 2002). Assuming the number of distinct genotypes for each population is finite, we can elicit a payoff matrix for the first population by simply determining the objective function values at each combination of genotypes with the opposing population's genotypes. Since the game is symmetric, the second population uses the transpose of this matrix.

### 2.3 MODEL ASSUMPTIONS

With an EGT model comes a variety of simplifying assumptions. In addition to infinite populations, assumptions must be made as to how individuals interact in order to obtain an assessment of fitness. Although many kinds of interactions are possible, we will retain the standard EGT assumption of *complete mixing* [1], meaning that during the evaluation phase of the algorithm, individuals in one population are assumed to have interacted with all members of the alternate population in pair–wise collaborations, and vice–versa. What this means, in terms of the algorithm, is that we will evaluate a given member of $x$ with *every* member of $y$ (a collaborator) for collaboration purposes, and take the average of the resulting fitness values.

We also retain the common assumption of proportional selection. However, it should be noted that the EGT framework can and does allow for analysis of coevolution using other selection methods (Ficici and Pollack, 2000a). Finally, although EGT models typically do not include variation operators, it has been recently noted that modeling variation is possible (Ficici and Pollack, 2000b), and that of course is the focus of this paper.

### 2.4 PROBLEM DOMAIN

Any two argument function can be mapped into a payoff matrix for a two population symmetric game (Wiegand et al., 2002). For simplicity our studies for this paper focus on a simple 2D parabolic function $f(x, y) = k - \left[(x_c - x)^2 + (y_c - y)^2\right]$. The domain values for both arguments are in the interval $[1, 8]$. Here the maximum value, $k = 33$ is found at the center of the bowl and is located at $(x_c, y_c) = (4, 4)$. In all cases each argument is represented as a 3-bit binary string implying that there are eight distinct genotypes represented for both populations, making a $8 \times 8$ payoff matrix.

We chose this landscape precisely because of its simplicity. While it is simple, we are still able to observe interesting effects variation has on search dynamics. Although in the past we have performed analysis on somewhat more complicated landscapes (Wiegand et al., 2002), the unimodality of the

[1] The word "mixing" here refers to how individuals interact, not variation. Unless it is otherwise stated, the reader should assume that "mixing" refers to variation in the remainder of this paper.

simple 2D parabolic function allows us to rule out certain, more complicated dynamical behaviors (such as convergence to suboptimal, local peaks) as a means of explaining some of these effects.

## 3 ANALYSIS WITHOUT VARIATION

Our goal is to better understand CCEA dynamics by analyzing the corresponding MPS models from a dynamic systems perspective. In particular, we are focusing on population trajectories, the existence and location of fixed points and their basins of attraction. In this section we summarize what is known for MPS models without variation, setting the stage for studying the effects due to variation.

### 3.1 STABILITY OF FIXED POINTS

If we exclude variation from our models, there is much which can be said analytically about MPS models. For example, any *strict* Nash equilibria must contain only pure strategies; that is, they must be at the basis vectors, the corners of the simplexes (Hofbauer and Sigmund, 1998). This means that in the absence of variational operators, we can expect the populations in our systems to converge to homogeneity. We also know, however, that mixed strategy equilibria are possible on the $bnd(\Delta^n \times \Delta^m)$ when the Nash points are not strict. This can happen when there are plateaus or ridges in the objective landscape, for instance.

It will be helpful for our discussion to provide some basic definitions. When populations $\vec{x}$ and $\vec{y}$ are both at basis vectors, we say that the those populations are *associated* or *correspond* with a payoff value $a_{i,j} \in A$, when $\vec{x} \cdot A\vec{y} = a_{i,j}$. This happens when $\vec{x}_i = 1$ and $\vec{y}_j = 1$, but all other components of both vectors are 0. Population vectors which are not basis vectors are said to be *polymorphic*. For simplicity of notation, we will use the term $\bar{v}$ to refer to a fixed point $(\vec{x}, \vec{y}) \in \Delta^n \times \Delta^m$, whether polymorphic or not.

To understand the fixed point limiting behaviors of these algorithms more fully, we present proofs of some useful properties about discrete time MPS models. Some of these properties are known to Evolutionary Biologists studying EGT, but since their focus is not on optimization we consolidate this information and offer theorems useful for the analysis of CCEAs in this paper. Moreover, the proofs of the above properties are very instructive. From it we are able to also show several things about when basis vector fixed points are purely stable and purely unstable, as well as provide the basis for understanding how certain kinds of local convergence problems can occur in a CCEA, even with infinite populations and no variation.

In order to provide some context for the proofs which follow it will be useful to outline our strategy for establishing this proof. The intent is to be able to make some general statements about the *Jacobian* of the system of equations generated by algebraic expansion of the replicator and selection equations evaluated at fixed points, $\bar{v}$ which are associated with the corners of the simplex. Since we can shuffle rows and columns of the payoff matrix and the game remains the same, without loss of generality we can restrict our discussion to the case where $\bar{v}$ corresponds to the element $a_{1,1}$ in the

$A$ payoff matrix, where $A$ can be expressed as shown below. All other cases follow as a result.

$$A = \begin{bmatrix} a_{1,1} & a_{1,2} & \cdots & a_{1,n} \\ a_{2,1} & \ddots & \cdots & a_{2,n} \\ \vdots & & & \vdots \\ a_{n,1} & \cdots & & a_{n,n} \end{bmatrix} \tag{5}$$

We begin by showing that there are particular patterns that will necessarily be found in the *Jacobian* when evaluated at the fixed point, $J_v(\bar{v}_{a_{1,1}})$. This structure allows us to make assertions about the what the eigenvalues of the *Jacobian* will be in the general case. These steps lead us directly to a theorem which tells us that the fixed point associated with the maximum value in the payoff matrix is always stable. We then provide several corollaries which follow directly from this theorem that give more information about the nature of stability and instability in basis vector fixed points.

**Lemma 1** *Let $J_v$ be the* Jacobian *of the system of equations generated by algebraic expansion of the two–population, $n$-strategy evolutionary game, where $A$ is the payoff matrix and the replicator equations are given in equations 1 – 4. Given the fixed point $\bar{v}_{a_{1,1}}$ associated with the $a_{1,1}$ item in $A$, all eigenvalues of the* Jacobian *evaluated at the $\bar{v}_{a_{1,1}}$ fixed point, $J_v(\bar{v}_{a_{1,1}})$, are the diagonal elements.*

**Proof:** We begin the proof by partitioning $J_v(\bar{v}_{a_{1,1}})$ into four partitions. We use these partitions to show that $J_v(\bar{v}_{a_{1,1}})$ must be in a particular form, then conclude it by proving that lemma 1 follows as a result of this form.

$J_v(\bar{v}_{a_{1,1}})$ can be partitioned into four equal sized partitions $B$, $C$, $D$, and $E$ shown in equation 6 below. It is clear by the definition of a *Jacobian* that $B$ will correspond to $\frac{\partial x'_i}{\partial x_j}$, $C$ to $\frac{\partial x'_i}{\partial y_j}$, $D$ to $\frac{\partial y'_i}{\partial x_j}$, and $E$ to $\frac{\partial y'_i}{\partial y_j}$. We will first restrict ourselves only to the $B$ partition.

$$J_v(\bar{v}_{a_{1,1}}) = \left[\begin{array}{c|c} B & C \\ \hline D & E \end{array}\right] \tag{6}$$

We omit the $n^{th}$ column and row from the matrix. This can be done because the $x_n$ variable may be re-written using the other components in $\vec{x}$ (that is, $x_n = 1 - \sum_{i=1}^{n-1} x_i$), and because the new proportion for the $x_n$ component in the next generation is fully specified by the equations without the redundant $x'_n$. Of course, a similar argument holds for the $n^{th}$ column and row of all four partitions.

The algebraic form of $u_i = a_{i,1}y_1 + a_{i,2}y_2 + \cdots + a_{i,(n-1)}y_{(n-1)} + a_{i,n}\left(1 - \sum_{i=1}^{n-1} y_i\right)$ can be obtained from equations 1 and 2. Since we know in the $B$ partition case that we will always be taking the partials with respect to one of the $\vec{x}$ variables, we can go ahead and substitute the $\vec{y}$ values from $\bar{v}_{a_{1,1}}$ since they will be considered constants in the derivatives anyway. This gives us a somewhat simpler algebraic form to use: $u_i = a_{i,1}y_1 + a_{i,n}(1 - y_1) = a_{i,1}$. We can also substitute most of the $\vec{x}$ values as constants as long as we can still legitimately take the partial derivatives. Let us *preserve* any elements that fall in the $i^{th}$ row or $j^{th}$ column, as well as those in the first column and row in order to examine the partial—all other values in $\vec{x}$ are zero so further simplification is possible. The

$x_i'$ algebraic form is shown below.

$$x_i' = \frac{a_{i,1}x_i}{x_1 a_{i,1} + x_i(a_{i,1} - a_{n,1}) + x_j(a_{j,1} - a_{n,1})}$$

If $i \neq j$ then we can substitute either 0 or 1 for $x_i$ when taking the partial with respect to $x_j$. If $i \neq 1$, then we know that $\frac{\partial x_i'}{\partial x_j} = 0$ since the numerator will remain a zero factor after the derivative. Therefore we can say that all elements in the $A$ partition of $J_v(\bar{v}_{a_{1,1}})$ are zero as long as $i \neq j$ and $i \neq 1$. The form of the partition is shown below in equation 7.

$$J_v(\bar{v}_{a_{1,1}}) = \left[ \begin{array}{cccc} j_{1,1} & j_{1,2} & \cdots & j_{1,n-1} \\ 0 & j_{2,2} & 0 & \cdots \\ \vdots & 0 & \ddots & 0 \end{array} \right. \tag{7}$$

By symmetry we know that the $E$ partition has the same form.

The $C$ and $D$ partitions never have $i = j$, so we know that their diagonals are always zero. There is only one case for these partitions that we need to worry about: when $i = 1$. Let us take the $i = 1$ case for $C$ and return to the replicator function to get the algebraic expression after appropriate constant substitutions $u_i = a_{i,1} + a_{i,j}y_j - a_{i,n}y_j$. After simplification, the selector equation is then $x_i' = \frac{x_i\left(a_{i,1}+a_{i,j}y_j-a_{i,n}y_j\right)}{x_1\left(a_{i,1}+a_{i,j}y_j-a_{i,n}y_j\right)}$. The terms in the numerator and denominator cross out when $i = 1$ and becomes a constant, the partial of which is zero. Thus all elements in the $C$ partition (and $D$ by symmetry) are zero.

Now that we have more knowledge about the structure of $J_v(\bar{v}_{a_{1,1}})$, let us look at obtaining the eigenvalues. Recall that to compute this, we must solve the characteristic equation such that $\det\left(J_v(\bar{v}_{a_{1,1}}) - \lambda I\right) = 0$. We can compute this by expansion of cofactors on the first column. From our above discussion we know that the first column of the *Jacobian* are all zeros except elements $j_{1,1}$.The cofactors of $j_{1,1}$ is the product of the diagonal terms $J_v(\bar{v}_{a_{1,1}})$ excluding $j_{1,1}$, of course. This can be seen by repeated application of the expansion of cofactors.

Thus the determinate is simply the product of the diagonal terms of the matrix $J_v(\bar{v}_{a_{1,1}}) - \lambda I$, so the roots of the characteristic equation will be the diagonal elements of the *Jacobian*. Therefore the eigenvalues of $J_v(\bar{v}_{a_{1,1}})$ are its diagonal elements. □

**Lemma 2** *Given $J_v(\bar{v}_{a_{1,1}})$, the following properties are true.*

$$\frac{\partial x_i'}{\partial x_i'}\left(\bar{v}_{a_{1,1}}\right) = \frac{a_{i,1}}{a_{1,1}} \; \forall i \neq 1 \tag{8}$$

$$= \frac{a_{n,1}}{a_{1,1}} \; i = 1 \tag{9}$$

$$\frac{\partial y_i'}{\partial y_i'}\left(\bar{v}_{a_{1,1}}\right) = \frac{a_{1,i}}{a_{1,1}} \; \forall i \neq 1 \tag{10}$$

$$= \frac{a_{1,n}}{a_{1,1}} \; i = 1 \tag{11}$$

(12)

**Proof:** Again let us focus on partition $B$ of $J_v(\bar{v}_{a_{1,1}})$. Returning to the replicator equation 4 after appropriate constant substitution, we get $u_i = a_{i,1}y_1 + a_{i,n}(1-y_1) = a_{i,1}$. There are two cases: $i \neq 1$ and $i = 1$. In the fist case, again retaining $x_i$ for the partial, we substitute the remaining values for $\vec{x}$ as constants and obtain the following selection equation and subsequent partial derivation.

$$\begin{aligned} x_i' &= \frac{x_i a_{i,1}}{x_1 a_{1,1} + x_i(a_{i,1} - a_{n,1})} \\ \frac{\partial x_i'}{\partial x_i'}\left(\bar{v}_{a_{1,1}}\right) &= \frac{a_{i,1} x_1 a_{1,1}}{\left(x_1 a_{1,1} + x_i a_{i,1} - x_i a_{n,1}\right)^2} \\ &= \frac{a_{i,1}}{a_{1,1}} \end{aligned}$$

In the case where $i = 1$ we do not preserve $x_i$ of course, so we obtain the following after relevant substitution.

$$\begin{aligned} x_i' &= \frac{x_1 a_{1,1}}{x_1 a_{1,1} + a_{n,1} - x_1 a_{n,1}} \\ \frac{\partial x_i'}{\partial x_i'}\left(\bar{v}_{a_{1,1}}\right) &= \frac{a_{1,1} a_{n,1}}{(a_{n,1} - a_{1,1} - a_{n,1})} \\ &= \frac{a_{n,1}}{a_{1,1}} \end{aligned}$$

The proof for the $\frac{\partial y}{\partial i}i$ case can be obtained from the $D$ partition by symmetry. $\square$

**Theorem 1** *Let $m$ be a unique maximum value in $A$, $m = \max(A)$. Given a non-polymorphic fixed point $\bar{v}$, if the payoff value $a_{i,j} = m$ and $\bar{v}$ corresponds with $a_{i,j}$, then $\bar{v}$ is a stable fixed point.*

**Proof:** Given $a_{i,j} = m$ and that $a_{i,j}$ can be moved into position $a_{1,1}$, then all eigenvalues are $< 1$ by lemma 2. $\square$

**Corollary 1** *Noting in lemma 2 that all the eigens for a particular $\bar{v}$ are ratios of $A$ values at some fixed column or row, and applying the same juxtaposition logic from theorem 1, it follows directly that any $a_{i,j}$ that is the unique maximum value of the $i^{th}$ row and $j^{th}$ column of $A$ is also a stable fixed point. More formally if*

$$\begin{aligned} a_{i,j} &> a_{l,j}\ \forall l \neq i, \ \ \textit{and} \\ a_{i,j} &> a_{i,k}\ \forall k \neq j, \end{aligned}$$

*then $\bar{v}_{a_{i,j}}$ is a stable fixed point.*

**Corollary 2** *As per corollary 1, lemma 2 and theorem 1 dictate that any $a_{i,j}$ that is the minimum value of the $i^{th}$ row and $j^{th}$ column of $A$ is a purely* unstable *fixed point (meaning* all *its eigenvalues are $> 1$). More formally if*

$$\begin{aligned} a_{i,j} &< a_{l,j}\ \forall l \neq i, \ \ \textit{and} \\ a_{i,j} &< a_{i,k}\ \forall k \neq j, \end{aligned}$$

*then $\bar{v}_{a_{i,j}}$ is a purely unstable fixed point. The global minimum value, $\min(A)$ is always purely unstable. Additionally any other fixed points at the corner of the simplex which are neither purely unstable nor stable will be an unstable saddle point.*

**Corollary 3** *From corollaries 1 and 2 we also know that the maximum number of stable basis vector fixed points is $n$ and the minimum number of stable basis vector fixed points is $1$. The same rule is also true for the number of purely unstable fixed points. Therefore the number of unstable saddle basis vector fixed points must be at least $n^2 - 2n$.*

What does this analysis tell us about convergence and optimization? First, let us say that by "convergence" here, we are loosely connecting our dynamical systems ideas to the intuitional notions of population convergence common in the EC community. When trajectories limit to a fixed point at the basis vector, this corresponds to the populations becoming homogeneous. The question is, where can this occur in terms of the fitness landscape? Recall that in our MPS model of cooperative coevolution, the payoff matrix is really just a quantized version of our fitness landscape. Given this, perhaps the most important thing worth noting is that (even with an infinite population, no variation, and unique values in the fitness landscape) a form of "*local*" convergence is possible, since trajectories can fall to basis vectors which correspond with suboptimal fitness values. This is not possible in the simple genetic algorithm under the same assumptions (Vose, 1999).

However, knowing the stability of a fixed point in a system does not necessarily indicate how likely it will be reached by any arbitrary initial condition, unless more is known about the dynamical system (Hofbauer and Sigmund, 1998). Fortunately, even without the deeper understanding required for formal proofs, considerable insight into these issues can be obtained by iterating the model computationally from randomly chosen initial conditions. In the following section we summarize these insights for MPS models without variation as a baseline for studying the effects of variation.

### 3.2 POPULATION TRAJECTORIES AND BASINS OF ATTRACTION

The basin of attraction of a given fixed point (or, indeed, any limiting behavior) is the set of initial points that will eventually map to that point (or appropriate limit behavior). Of interest is the size of a fixed point's basin of attraction relative to the other fixed points to which trajectories go. In other words, viewing convergence to a basis vector as a collapse to homogeneity in both populations, knowing the relative sizes of the basins of attraction of each possible pair of homogeneous populations should give us insight into *how likely it is that a pair of random initial populations will converge to some particular pair of homogeneous populations*.

There is reason to believe that the size of the basins of attraction of a fixed point indicated by a basis vector has more to do with relative local column and row values in the payoff matrix than how large the specific payoff value is at that point. In other words, there is reason to believe that broad, suboptimal peaks will pull trajectories away from taller, more narrow peaks. This is a form of local convergence that has been recognized in CEAs (Ficici and Pollack, 2000b; Wiegand et al., 2002).

To estimate the size of a basin of attraction we use a simple method called a *rain gauge measure* (Alligood et al., 1996). Details motivating this technique for the MPS model are provided by Wiegand et al. (2002). Briefly, this method consists of generating a large number of (uniformily) randomly generated initial conditions and, for each starting point, iterating the model until a fixed point is reached. The percentage of trajectories leading to a particular fixed point is used as an estimate of the size of its basin of attraction.

So, for example, if we apply this technique to the simple unimodal quadratic function described in section 2.4, we obtain the expected result that, for MPS models with no variation, every trajectory leads to the basis vector fixed point associated with the global optimum. That is, when we choose an initial starting state for the population at random, the model predicts that a CCEA algorithm will converge to homogeneity at the global maximum of this function.

Iterating the model not only affords us with the opportunity to assess the relative sizes of the basins of attraction. An additional benefit is its ability to help us to visualize and characterize the trajectories themselves. We do this using a plot similar to the so-called *takeover curves* used for standard GA analysis (Goldberg and Deb, 1990), except in our case these curves must be two dimensional in order to capture both populations. The plots are constructed by first identifying which genotypes correspond with the maximum payoff value in each population and plotting over time the proportions of these genotypes in their respective $\vec{x}$ and $\vec{y}$ population vectors. Figure 1 illustrates this for several example trajectories. Each curve begins at the point indicated by a hollow circle, and terminates at the point indicated by an "x" (in this case all of these terminate in the corner). Every 100 steps of the trajectory are marked on the curves to get an idea of the rate of progress of the curves, though all points on the plot represent steps produced by the model. In this way, one can track the proportions of the components associated with the maximum value over time as they move from the initial population configurations toward its ultimate limit behavior. Asymmetries in the projected trajectories are due to asymmetric differences in the initial points, which were chosen randomly, or asymmetries which exist in the function itself (the latter of which, in this case, is minimal).

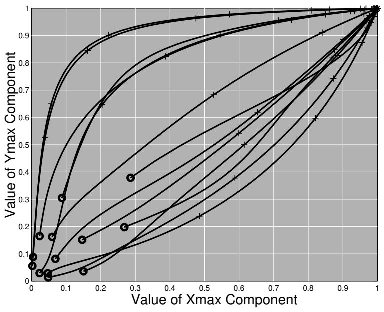


**Figure 1** 2D Takeover plot for trajectories operating on the simple quadratic problem with no variation. Curves begin at points indicated by hollow circles, and terminates at the point indicated by an "x". Every 100 steps of the trajectories are marked.

Using these methods it is fairly easy to show, for example, that MPS models without variation can and do converge to local suboptimal peaks on multi-peak landscapes (Wiegand et al., 2002). This should already be clear from our earlier analysis; however, since we now know that suboptimal values in the payoff matrix that are strictly maximal on the its row and column are associated with basis vectors which are stable fixed points of the system. Using this knowledge, it isn't difficult to construct problems which are likely to "pull" trajectories from the global peak. We do this by making sure the rows and columns in the payoff matrix corresponding with the suboptimal local peak are significantly biased over the row and column values corresponding with the global peak. Indeed, this is precisely what the Wiegand et al. (2002) study does, and the model validation results support this hypothesis.

## 4 ANALYSIS WITH VARIATION

In order to extend the MPS framework to model variational operators, we have employed the methods outlined by (Vose, 1999) in which the dynamical system becomes a composition of the orignal model and a variational mixing function, $\mathcal{M}$. We assume that the variational operators are the same for both populations and the populations have the same number of distinct genotypes, so the same $\mathcal{M}$ can be used for both populations. With these asumptions our MPS model now becomes: $\mathcal{G}_x = \mathcal{M}\left(\mathcal{S}\left(\mathcal{F}_x, \vec{x}\right)\right)$ and $\mathcal{G}_y = \mathcal{M}\left(\mathcal{S}\left(\mathcal{F}_y, \vec{y}\right)\right)$.

By adopting this approach we also commit ourselves to modeling populations of binary strings and the variation due to the standard binary string operators of crossover and mutation. The mixing function works by first eliciting the probabilities that the all zero string is produced given two parents of any genotype. These probabilities form an $n \times n$ matrix, $M$. The population vectors resulting from selection are then processed using permutations of this matrix (see Vose (1999) for more details). Given this and the $\vec{x}'$ and $\vec{y}'$ results from the standard MPS replicator Equations 1- 4, we can now obtain the next generation's population states (now notated $\vec{x}''$ and $\vec{y}''$) from the following equations:

$$x''_k = \sum_{i,j \in \Omega} x'_i x'_j M_{i \oplus k, j \oplus k} \tag{13}$$

$$y''_k = \sum_{i,j \in \Omega} y'_i y'_j M_{i \oplus k, j \oplus k} \tag{14}$$

The reader is referred to Vose (1999) for a more in depth explanation of how these mixing matrices are constructed for particular operators.

### 4.1 FIXED POINTS UNDER MIXING

There is little doubt that variational operators significantly change the underlying dynamical system of a CCEA algorithm, just as they do in a traditional EA. Not only can the underlying limiting properties of the fixed points of the system change, but the location of the fixed points themselves can change. To see this, note that for a fixed point of the MPS model to *also* be a fixed point of the variational model, $\vec{x}'' = \vec{x}' = \vec{x}$ and $\vec{y}'' = \vec{y}' = \vec{y}$, which means we can re-write Equations 13 and 14 as follows.

$$x''_k = \sum_{i,j \in \Omega} x_i x_j M_{i \oplus k, j \oplus k} \tag{15}$$

$$y''_k = \sum_{i,j \in \Omega} y_i y_j M_{i \oplus k, j \oplus k} \tag{16}$$

As we saw earlier, the fixed points of MPS models without variation were basis vectors. What happens when variation is added? Suppose we are interested in the basis vectors associated with payoff value $a_{pq}$, that is $x_i = 0, \forall i \neq p, x_p = 1$ and $y_i = 0, \forall i \neq q, y_q = 1$. If such is the case, then the resulting values from the above equations is always 0, except when $i = j = p$ in the first case and $i = j = q$ for the second case. Thus, the next point in the trajectory, when starting at the basis vector is the following:

$$x''_k = M_{p \oplus k, p \oplus k} \tag{17}$$

$$y''_k = M_{q \oplus k, q \oplus k} \tag{18}$$

There are two important things to note about this simplification. First, one can ascertain the $\mathcal{G}_x^1(\vec{x},\vec{y}), \mathcal{G}_y^1(\vec{x},\vec{y})$ step of the MPS model from any arbitrary basis vector point *from just the mixing matrix*. Second, resulting values turn out to be diagonals of the $p^{th}$ and $q^{th}$ permutation of the mixing matrix for $\mathcal{G}_x$ and $\mathcal{G}_y$, respectively. If this diagonal is equal to the original bases, then the fixed point of the original system is also a fixed point of the system under variation, and if not then it is not. As we will discuss in the following sections, this is always true for crossover and never true for mutation.

Assessing fixed point stability under mixing is harder, since the fixed points may now be in the interior of the simplex product. This means that it becomes necessary to simultaneously solve the collective system for $\mathcal{G}_x(\vec{x},\vec{y}) = \vec{x}$ and $\mathcal{G}_y(\vec{x},\vec{y}) = \vec{y}$. However, when we are certain that the basis vectors are fixed points, even under mixing, we can evaluate them as we did in the previous paragraph.

### 4.2 STABILITY OF FIXED POINTS

Unlike the MPS models without variation, the stability of fixed points of the model under variation is a function of $M$. Moreover, this dependence is not due simply to the inclusion of $M$ itself into the model, but also from the resulting non-linearity added from the crossover operation itself. The proof of this is trivial, but we offer it for the sake of completeness.

**Theorem 2** *Let $\bar{v}$ be a fixed point of the cooperative coevolutionary algorithm with variation as described by equations 13 and 14. The stability of such a fixed point will depend on the specific values in the mixing matrix, $M$.*

**Proof:** To assess the stability of a fixed point, we must first know what the fixed point is. We will see shortly that while some of the fixed points are known for crossover, this is not necessarily true for mutation (or for mutation *and* crossover). Still, assuming we knew what the fixed point was, we could take the *Jacobian* of the system evaluated at that fixed point. Let's look at one term $\frac{\partial x_k''}{\partial x_l}$, but first let's expand and re-write the equations 13 and 14. We use the variable $f$ for the convenience of notational simplification only.

$$
\begin{aligned}
x_k'' &= \sum_{i,j\in\Omega} \frac{(A\vec{y})_i x_i (A\vec{y})_j x_j M_{i\oplus k, j\oplus k}}{(\vec{x}^T A\vec{y})^2} \\
&= \frac{\sum_{i,j\in\Omega} (A\vec{y})_j x_i (A\vec{y})_j x_j M_{i\oplus k, j\oplus k}}{(\vec{x}^T A\vec{y})^2} = \frac{f}{(\vec{x}^T A\vec{y})^2} \\
\frac{\partial x_k''}{\partial x_l} &= \frac{\frac{\partial f}{\partial x_l}\cdot\left(\vec{x}^T A\vec{y}\right)^2 - f\cdot\frac{\partial}{\partial x_l}\left(\vec{x}^T A\vec{y}\right)^2}{(\vec{x}^T A\vec{y})^4} \\
&= \frac{\frac{\partial f}{\partial x_l}\cdot\left(\vec{x}^T A\vec{y}\right)^2 - f\cdot 2\left(\vec{x}^T A\vec{y}\right)(A\vec{y})_l}{(\vec{x}^T A\vec{y})^4}
\end{aligned}
$$

We can take the partial of $f$ now and obtain $(A\vec{y})_l \sum_{i\in\Omega, i\neq l} (A\vec{y})_i x_l M_{i\oplus k, l\oplus k} + 2(A\vec{y})_l^2 x_l M_{l\oplus k, l\oplus k}$. By symmetry, the same is true for $\frac{\partial y_k''}{\partial y_l}$. From this we learn that both terms in the numerator depend on values from $M$, and that they cannot be eliminated. $\square$

Although this fact was perhaps already obvious, the proof is instructive since it illustrates the fact that the non-linearity introduced by equations 13 and 14 presents a far more complicated expression that requires the change in any given component depends on all the components from *both* populations. This differs from the model without variation, since in that case taking the derivative allowed us to eliminate all components of one population from the expression.

### 4.3 PARAMETERIZED UNIFORM CROSSOVER

We now focus on the effects of crossover alone. In this case, regardless of the type of crossover, the probability of obtaining some string $k$ when crossing over two identical parents is 0 unless the parents are themselves $k$. When $k$ is crossed over with $k$, the probability that the resulting child is also $k$ is 1. Thus the diagonal of the $k^{th}$ permutation of $M$ is always 1 at $m_{kk}$ and zero everywhere else, implying that the pairs of basis vectors forming fixed points of the MPS model without variation are still fixed points under crossover. This is consistent with our intuitional understanding of the effect of crossover on a totally homogeneous population (namely that there is no effect).

In order to study the effects of crossover on population trajectories and basins of attraction, we need to complete our MPS model by selecting a particular crossover operator and constructing the corresponding mixing matrix $M$. For our initial work we chose to model parameterized uniform crossover (Syswerda, 1989; Spears and De Jong, 1991). Here $p_c$ represents the probability that a crossover event will occur. If a crossover event occurs, two individuals are mated and each bit position is considered independently for potential exchange. The parameter $p_s$ represents the probability that the values at a given bit position will be exchanged between the mates. Therefore $p_s = 0.5$ corresponds with traditional uniform crossover.

Having done so, we are now in a position to repeatedly iterate the model as before on the same simple two-dimensional parabola and assess the effects of crossover on population trajectories and basins of attraction. Since we know that the basis vectors of the Cartesian product of the unit simplexes are also fixed points of this model (though not necessarily stable in the same places), we can perform similar kinds of rain-gauge measures as was done in the previous section. However, we must be careful to account for any trajectories which do not converge to a basis vector, since the existence of interior fixed points have yet to be formally eliminated as a possiblity as it was without variation. As it turned out, all of the trajectories converged to a basis vector of one sort or another. Moreover, existing analysis suggests to us that *stable* interior equilibria are not possible in the simple GA, with only crossover (Vose, 1999).

Initially, we set $p_s = 0.5$ to obtain pure uniform crossover and varied the rate of crossover, $p_c$ between 0.0 and 1.0. All trajectories moved to one of four basis vectors, those vectors associated with the $a_{4,4}$, $a_{4,5}$, $a_{5,4}$, and $a_{5,5}$ payoff values (the center four cells of the payoff matrix, from the top left as described in equation 5 and page 6). Example results are shown in Table 1. This shows the percentage of trajectories from the randomly chosen initial populations that go to each of the four afore mentioned basis vectors. More specifically, it shows that as the rate of crossover is increased, the number of initial points which eventually converge to a fixed point associated with the global maximum shrinks. In other words, the measure of the size of the basin of attraction of that fixed point associated with the global peak is reduced by increasing crossover.

The reader should be careful not to draw too many conclusions about the significance of the ”nearness“ of these fixed points from a topological point of view. From the simplex product space, every basis vector is a distance of either 1.0 or $\sqrt{2}$ from every other basis vector, of course. Moreover, as long as the relationships between the strategies of the game remain the same, one could permute the payoff matrix and the underlying dynamics would not change at all. Thus it is possible to imagine the very same model with resulting measures that seem topologically farther apart. The relationship almost certainly has to do with the level of fitnes on a given row and column relative to other strategies.

**Table 1** Rain gauge results of model validation studies on MPS cooperative coevolution model with uniform crossover. The tables represent the measure of the basins of attraction of the fixed points associated with the $a_{4,4}$, $a_{4,5}$, $a_{5,4}$, and $a_{5,5}$ payoff values. The probability of crossover, $p_c$ is varied.

| $p_c = 0.00$ | | |
|---|---|---|
| | 100 % | 0 % |
| | 0 % | 0 % |

| $p_c = 0.10$ | | |
|---|---|---|
| | 81.7 % | 11.7 % |
| | 6.6 % | 0 % |

| $p_c = 0.20$ | | |
|---|---|---|
| | 75 % | 15.8 % |
| | 8.4 % | 0.8 % |

| $p_c = 1.00$ | | |
|---|---|---|
| | 57.5 % | 23.3 % |
| | 11.7 % | 7.5 % |

We also looked at parameterized uniform crossover when $p_s = 0.2$ and $p_c = 1.0$. The results for this showed that the $(4,4)$ basin captured roughly 62.6% of the trajectories, while the $(4,5)$, $(5,4)$, and $(5,5)$ basins captured 16.2%, 17.0% and 4.2% of the trajecories, respectively. Again, all trajectories found their way to one of these four basis vectors.

This is an interesting result. This suggests that it is possible that previously unstable basis vector fixed points become stable attractors under crossover. In a sense, crossover seems to "distract" trajectories from always converging to the basis vector associated with the maximum value. To get a sense for why this might be, we looked at the 2D takeover plots shown in figure 2.

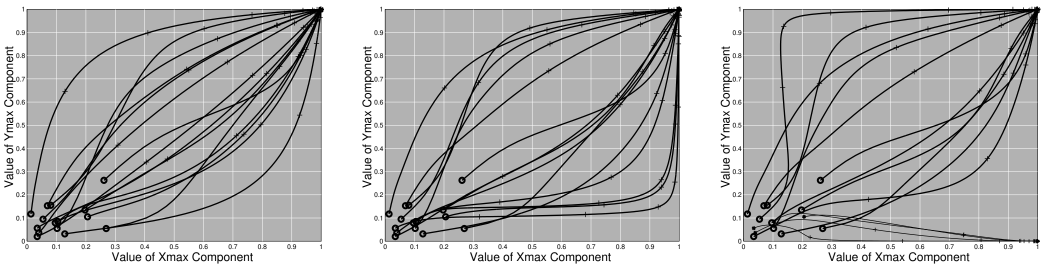


**Figure 2** 2D takeover plots for MPS trajectories in with uniform crossover. Reading from the left to the right, $p_c = 0.00$, 0.05, and 0.10.

There is clearly some kind of stretching transformation going on as a result of crossover. As the rate of crossover increases, the trajectories are pushed away from the center and move toward the edges at a much faster rate. In some cases these trajectories are drawn to basis vector fixed points (homogeneous populations) not associated with the global optimum. This behavior is observed in CCEA applications as well where one population converges much faster than the other and reduces the dimensionality of the search by collapsing the space in which trajectories can pass to a face of the Cartesian product simplex. This corresponds to a reduction of the space to the unit simplex for the second population, still evolving population.

We believe this happens due to the accelerating effect crossover can have on population convergence (Menon, 2002; Rabani et al., 1998). This acceleration is applied asymmetrically, since the initial conditions of the populations are almost certainly asymmetric. To test this hypothesis, we ran an additional experiment in which all initial points for $\vec{x}$ were chosen uniformly at random from the unit simplex, and all initial points for $\vec{y}$ were set symmetrically, $\forall i \; (\vec{y})_i = (\vec{x})_i$, where $(\vec{x})_i$ is the $i^{th}$ initial point. Although not all trajectories converge to proper basis vector (the corner associated with the maximum value), *none* converged sooner in one population than in the other.

### 4.4 BIT-FLIP MUTATION

In a similar fashion we can construct the mixing matrix M for the bit-flip mutation operator. In this case, the diagonal elements of $M$ cannot be basis vectors as long as $p_m \neq 0$ and $p_m \neq 1$ since a population which is completely converged cannot remain so after mutation (in an infinite population model). Therefore it is fair to conclude that the basis vectors are no longer fixed points under mutation.

Unfortunately, this means that rain gauge measures of the type we have been performing are not reasonable. Instead, we can measure the distance of the point to the basis vector associated with the maximum payoff value. We ran trajectory studies with $p_m$ set to a variety of values. The result was that all the trajectories converged on the same fixed point in all cases for a given $p_m$ value, but that this point moved into the interior of the simplex product as mutation was increased. Figure 3 shows the 2D takeover plots for a few of these runs, which allow us to visualize only a *projection* of the trajectories. Still we notice that the distortions of the trajectories due to mutation are quite different than those produced by crossover.

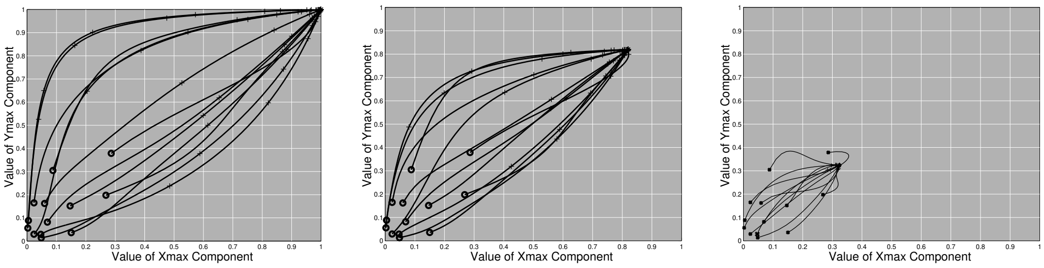


**Figure 3** 2D takeover plots for MPS trajectories in with bit-flip mutation. Reading from the left to the right, $p_m = 0.00$, 0.005, and 0.05.

This behavior matches our intuition, as well as known results for the simple GA. As mutation is increased, the limiting behavior is characterized as a distribution of population states which spread out away from the peak into the rest of the fitness landscape.

### 4.5 MUTATION AND CROSSOVER

Studying the effects of MPS models with both crossover and mutation is now a straightforward exercise in producing a combined mixing matrix. It is clear that the basis vectors are no longer fixed points. What is unclear is whether the combination of both operators will amplify the trajectory distortions produced individually, diminish them, or produce some other effect.

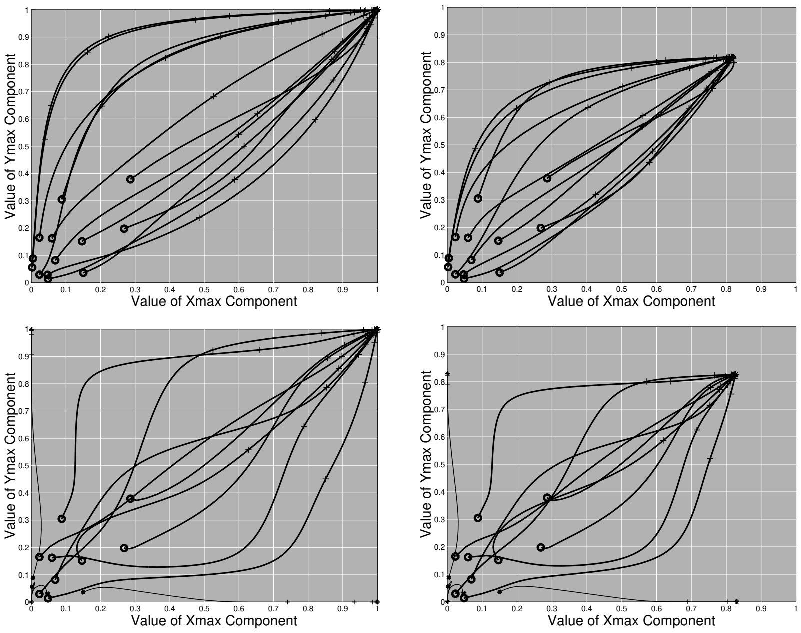


**Figure 4** 2D takeover plots for MPS trajectories in with bit-flip mutation and parameterized uniform crossover. Reading from the top left plot clock-wise, $(p_c = 0, p_m = 0)$, $(p_c = 0, p_m = 0.005)$, $(p_c = 1.0, p_m = 0.005)$, and $(p_c = 1.0, p_m = 0)$. In all cases with crossover, $p_s = 0.2$.

Figure 4 shows four combinations of the model. Interestingly, the result of combining the variation operators is additive in a sense. In our studies, trajectories fall towards the basis vectors associated with the center four payoff values as they did under crossover alone. The bottom two graphs in figure 4 shows us a projected version of this effect. However, observing the right two graphs, we see that the limit point is pulled into the interior of the simplex product space due to mutation.

# 5 CONCLUSIONS

The goal of this paper was to understand better the dynamical properties of CCEAs particularly with respect to their use as function optimizers. The approach taken was to extend existing MPS EGT models to include standard crossover and mutation operators, and relate the fixed points of these models to local/global optima convergence properties. This analysis was based on both theorems concerning model properties and the use of rain gauge measures and trajectory visualization techniques.

The result is a much clearer understanding of how the variation due to crossover and mutation affects the dynamics of CCEAs and their ability to converge to local/global optima. These results are clearly an intermediate step to a full understanding of these issues. The rain gauge measures and takeover plots provide useful intuitions rather than formal proofs. The MPS models still have a number of simplifying assumptions such as complete mixing and infinite populations that need to be weakened or eliminated in order to make stronger predictions about practical CCEA applications. However, we are pleased in general with the usefulness of the EGT framework as a tool for analyzing coevolutionary systems and plan further work in this area. Our next steps include continued formal analysis of variation, as well as the introduction of Markov modeling methods to relax the infinite population assumption. Afterwards, we would like to develop some probability-based models which effectively bias the population vectors based in ways that might be analogous to existing partial mixing models.

## References


Alligood, K., T. Sauer, and J. Yorke (1996). *Chaos: An Introduction to Dynamical Systems*. Springer–Verlag.

Ficici, S. and J. Pollack (2000a). Game–theoretic investigation of selection methods used in evolutionary algorithms. In D. Whitley (Ed.), *Proceedings of CEC 2000*, pp. 880–887. IEEE Press.

Ficici, S. and J. Pollack (2000b). A game-theoretic approach to the simple coevolutionary algorithm. In M. Schoenauer, K. Deb, G. Rudolph, X. Yao, E. Lutton, J. Merelo, and H.-P. Schwefel (Eds.), *Proceedings of the Sixth International Conference on Parallel Problem Solving from Nature (PPSN VI)*, pp. 467–476. Springer-Verlag.

Goldberg, D. and K. Deb (1990). A comparative analysis of selection schemes used in genetic algorithms. In G. Rawlins (Ed.), *Foundations of Genetic Algorithms (FOGA)*, pp. 69–93. Morgan Kaufmann.

Hofbauer, J. and K. Sigmund (1998). *Evolutionary Games and Population Dynamics*. Cambridge University Press.

Maynard-Smith, J. (1982). *Evolution and the Theory of Games*. Cambridge University Press.

Menon, A. (2002). The point of point crossover: Shuffling to randomness. In L. Spector (Ed.), *Proceedings of the Genetic and Evolutionary Computation Conference (GECCO) 2002*, pp. 463–471. Morgan Kaufmann.

Potter, M. and K. De Jong (1994). A cooperative coevolutionary approach to function optimization. In Y. Davidor and H.-P. Schwefel (Eds.), *Proceedings of the Third International Conference on Parallel Problem Solving from Nature (PPSN III)*, pp. 249–257. Springer-Verlag.

Potter, M. and K. De Jong (2000). Cooperative coevolution: An architecture for evolving coadapted subcomponents. *Evolutionary Computation 8*(1), 1–29.

Rabani, Y., Y. Rabinovich, and A. Sinclair (1998). A computational view of population genetic. *Random Structures and Algorithms 12*, 313–334.

Spears, W. and K. De Jong (1991). On the virtues of parameterized uniform crossover. In R. Belew and L. Booker (Eds.), *Proceedings of the Fourth International Conference on Genetic Algorithms (ICGA)*, pp. 230–236. Morgan Kaufmann.

Syswerda, G. (1989). Uniform crossover in genetic algorithms. In J. D. Schaffer (Ed.), *Proceedings of the Third International Conference on Genetic Algorithms (ICGA)*, pp. 2–9. Morgan Kaufmann.

Vose, M. (1999). *The Simple Genetic Algorithm*. MIT Press.

Weibull, J. (1992). *Evolutionary game theory*. Cambridge, MA: MIT Press.

Wiegand, R. P., W. Liles, and K. De Jong (2002). Analyzing cooperative coevolution with evolutionary game theory. In D. Fogel (Ed.), *Proceedings of CEC 2002*, pp. 1600–1605. IEEE Press.