Focusing versus Intransitivity
Geometrical Aspects of Co-evolution

Anthony Bucci and Jordan B. Pollack

Brandeis University, Waltham MA 02454, USA
{abucci*,pollack}@cs.brandeis.edu
http://demo.cs.brandeis.edu/

Abstract. Recently, a minimal domain dubbed *the numbers game* has been proposed to illustrate well-known issues in co-evolutionary dynamics. The domain permits controlled introduction of features like intransitivity, allowing researchers to understand failings of a co-evolutionary algorithm in terms of the domain. In this paper, we show theoretically that a large class of co-evolution problems closely resemble this minimal domain. In particular, all the problems in this class can be embedded into an ordered, *n*-dimensional Euclidean space, and so can be construed as greater-than games. Thus, conclusions derived using the numbers game are more widely applicable than might be presumed. In light of this observation, we present a simple algorithm aimed at remedying focusing problems and relativism in the numbers game. With it we show empirically that, contrary to expectations, focusing in *transitive* games can be more troublesome for co-evolutionary algorithms than intransitivity. Practitioners should therefore be just as wary of focusing issues in application domains.

1 Introduction

[1] discusses a minimal substrate which can be used to illustrate several issues plaguing co-evolutionary dynamics. Individuals in this substrate are simply tuples of numbers. Because of the simplicity of the individuals, we are able to see more clearly what goes wrong when an algorithm fails to work as hoped. The authors explored two variants of the numbers game, a transitive and an intransitive one. The intransitive numbers game proved problematic for a conventional, fitness proportionate co-evolutionary algorithm. One issue which arose in the numbers game experiments which is of particular interest to us is the problem of overspecialization. Individuals had multiple dimensions on which they could vary. It was observed that some individuals would *focus* on one dimension at the expense of another. We will refer to this issue as *the focusing problem*. A detailed discussion of this problem can be found in [2].

In this paper, we will show that a large class of co-evolutionary domains, even intransitive ones, can be viewed as *n*-dimensional, transitive numbers games, for some unknown dimension *n*. This observation raises an important question: where have the intransitivities of the original domain gone? As we will see, the mathematics of *Pareto co-evolution* [3],[4] turns intransitive cycles into sets of non-dominated individuals [5]; what remains then are the transitive relations among individuals.

* Corresponding author

Our main mathematical tool will be the partial order decomposition theorem. This theorem states that any finite partially ordered set (poset) can be decomposed into an intersection of n linear orders, for some number n called the *partial order dimension*. This dimension n is meaningful geometrically: one corollary of the theorem is that an n-dimensional poset can be embedded monotonically into $\mathbb{R}^n$. As shown in [5], co-evolution problems expressible with a payoff function $p : S \times T \to R$ (R preordered) can be regarded as preorderings on the set S of candidate solutions. Modulo technicalities caused by preordering, such co-evolution problems can therefore also be viewed as suborders of $\mathbb{R}^n$. Or, to put it simply: as n-dimensional transitive numbers games.

While this result is purely theoretical, our empirical results suggest that we may reasonably approximate the mathematics in algorithms, at least for the problems we have tried. In other words, a naïve implementation of Pareto co-evolution can effectively solve intransitive problems.

Surprisingly, we also find that a co-evolutionary algorithm with a diversity maintenance mechanism can tackle the intransitive numbers game, similarly calling into question the importance of intransitivity in assessing co-evolutionary algorithm failure. Indeed, with a *transitive* variant of the numbers game designed to emphasize the focusing problem, co-evolution with diversity maintenance fails, and the full power of Pareto co-evolution is required to solve the problem. The conclusion we draw from these results is that focusing, rather than intransitivity, is the more important domain feature to consider in some applications.

We will be considering co-evolution problems which can be expressed with a payoff function of form $p : S \times T \to R$, where R is assumed to be partially ordered. For simplicity we will assume S, T and R are finite sets; though this assumption is not strictly necessary, it does simplify arguments.[1] We make no further assumptions about structure on R. This set might consist of numbers, or it might contain symbolic values like *lose* and *win*.

This paper is organized as follows. In section 2, we set up and state the partial order decomposition theorem. Making use of the mathematical notation and terminology established in [5], we apply this theorem to co-evolution problems and prove and important corollary, the preorder decomposition theorem, which makes up the backbone of our claim that co-evolution problems have significant geometric aspects. In section 3 we describe our experiments.

2 Orders and Co-evolution

In this section we set up and state the poset decomposition theorem. Our presentation borrows from [6], which should be consulted for details and proofs. We next prove a corollary which we call the preorder decomposition theorem. Finally, we apply our results to co-evolution problems of form $p : S \times T \to R$, showing that any such problem can be embedded into Euclidean n-space for some unknown n. Finally, we work through an example to illustrate the concepts.

[1] What we really need for most results is that the induced preorder on S be finite-dimensional, which is true of many infinite preorders too.

2.1 Poset Decomposition

We will write $\leq$ for orders. We may subscript like $\leq_R$ to emphasize we are in the poset R. Recall that a *monotone* function between two posets Q and R is a function $f : Q \to R$ such that whenever $q_1 \leq_Q q_2$, then $f(q_1) \leq_R f(q_2)$. Such a function thus preserves order relations. Recall also that f is *injective* if, whenever $q_1 \neq q_2$ then $f(q_1) \neq f(q_2)$. f is an *embedding of Q into R* if it is both monotone and injective. An embedding f essentially realizes Q as being part of R.

In this section, assume all posets are finite. We begin with the notion of linear extension:

Definition 1 (Linear Extension). *A* linear extension *of R is a total ordering L of the elements of R which is consistent with R's order. In other words, if S is the underlying set of both R and L, the identity function $\mathbf{1}_S : S \to S$ is monotone with respect to R in the domain and L in the range.*

Example 2. Let R have elements $\{a,b,c\}$ and relations $a \leq c$, $b \leq c$ (i.e., a and b are incomparable). Then one linear extension of R puts these elements in order $a \leq b \leq c$; call it L_1. R has a second linear extension L_2 putting the elements in order $b \leq a \leq c$. In light of this example, we have the following:

Definition 3 (Linear Realizer). *A* linear realizer *of a (finite) poset R is a set $\{L_1,\ldots,L_n\}$ of linear extensions of R such that $\bigcap_{i=1}^{n} L_i = R$. The intersection means that the only comparisons in $\bigcap_{i=1}^{n} L_i$ are the ones which are in all the L_i; all other pairs of elements are incomparable.*

Example 4. In example 2, L_1 and L_2 constitute a linear realizer $\{L_1,L_2\}$ of R. To see this, notice that $a \leq c$ and $b \leq c$ in both L_1 and L_2, whereas $a \leq b$ in L_1 while $b \leq a$ in L_2. Thus, in $L_1 \bigcap L_2$, $a \leq c$ and $b \leq c$ whereas a and b are incomparable. These relations are exactly the ones in R; hence, $R = L_1 \bigcap L_2$.

We have been leading up to the following fundamental fact about posets which we state without proof.[2]

Theorem 5 (Poset Decomposition Theorem). *Every finite poset has a minimal realizer.*

The "minimal" in the name "minimal realizer" means the linear realizer contains a minimum number of linear extensions. This minimum, call it n, is the *dimension* of R; alternately, R is called an n-dimensional poset. For instance, the poset in examples 2 and 4 is two-dimensional. The justification for using the word "dimension" is via the following two lemmas:

Lemma 6. *Every linear extension L of R gives rise to an embedding $x : R \to \mathbb{N}$.*

[2] See [6] for details. The crux of the proof is to show that R has at least one finite linear realizer.

Proof. A linear extension of R is essentially a choice for putting the elements of R into a list. If $R = \{s_1, \dots, s_m\}$, then L will be $s_{\sigma(1)} \le s_{\sigma(2)} \le \dots \le s_{\sigma(m)}$, σ being some permutation of $1, \dots, m$. Let us reindex R by defining $t_i = s_{\sigma(i)}$. Then, the mapping $x : R \to \mathbb{N}$ defined by $t_i \mapsto i$, is monotonic by construction. It is also injective, since we only index distinct elements of R. □

Lemma 7 (Embedding Lemma). *Every linear realizer* $\{L_1, \dots, L_n\}$ *of* R *gives rise to an embedding* $\phi : R \to \mathbb{N}^n$.

Proof. By lemma 6, each L_i gives rise to an injective, monotone function $x_i : R \to \mathbb{N}$. These functions define a sort of coordinate system for R. Define the map $\phi : R \to \mathbb{N}^n$ by $s \hookrightarrow (x_1(s), \dots, x_n(s))$ for all $s \in R$. Each coordinate x_i of ϕ is injective and monotone; thus ϕ itself is too. □

Remark 8. In particular, if R is an n-dimensional poset, it has a minimal realizer $\{L_1, \dots, L_n\}$. By lemma 7, there is thus an embedding $\phi : R \to \mathbb{N}^n$. $\mathbb{N}^n$ embeds into $\mathbb{R}^n$, and so we see that ϕ can be regarded as embedding R into ordered, n-dimensional Euclidean space. n is minimal in this case, so R *cannot* be embedded into m-dimensional space for some smaller m. Thus the name "n-dimensional poset."

2.2 Applications to Co-evolution

To complete the picture, we need to see how to apply the results of the previous section to co-evolution problems. First, let us recall some important definitions and notation from [5]. For any function $f : S \times T \to R$, where S is a set and R is a poset, write S_f for the preordering induced on S by pullback, and write the order on S_f as $\le_f$. This definition means that $s_1 \le_f s_2$ exactly when $f(s_1) \le_R f(s_2)$.[3]

Any function of form $S \times T \to R$ can be curried to a function of form $S \to [T \to R]$, where $[T \to R]$ stands for the set of all functions from T to R. If $p : S \times T \to R$, write $\lambda t.p$ for the corresponding curried function.

Consequently, starting from a co-evolution problem $p : S \times T \to R$, with R a poset, there is a corresponding *preorder* structure on the set S, namely $S_{\lambda t.p}$. The basic idea is that two candidate solutions s_1 and s_2 lie in the relation $s_1 \le_{\lambda t.p} s_2$ exactly when s_2's array of outcomes *covers* s_1's. In other words, s_2 does at least as well as s_1 does against every possible opponent. We refer the reader to [5] for details and examples.

The poset decomposition theorem applies to partial orders, not preorders, so we need to adjust it slightly. Our approach is to observe that every preorder R comes with an equivalence relation defined: $s_1 \sim s_2$ if and only if $s_1 \le_R s_2$ and $s_2 \le_R s_1$. One way to think about this relation is in the context of an objective function $f : S \to \mathbb{R}$. $s_1 \sim s_2$ exactly when the individuals s_1 and s_2 have the same fitness. In a multi-objective context, $s_1 \sim s_2$ when s_1 and s_2 have the same objective vector. Given this equivalence relation, we can then prove the following:

Lemma 9. *Let R be a (finite) preorder. There is a canonical partial order $Q = R/\sim$ and a surjective, monotone function $\pi : R \to Q$ (called the* projection*) such that $\pi(s_1) = \pi(s_2)$ if and only if $s_1 \sim s_2$, for all $s_1, s_2 \in R$. Q is called the* quotient *of R.*

---

[3] We can interpret this definition in terms of fitness functions: $s_1 \le_f s_2$ if s_2's fitness is at least as high as s_1's.

Proof. The proof that Q is a well-defined partial order can be found in [5]. Let us show that π is surjective and monotone. Define $\pi : R \to Q$ by $\pi(s) = [s]$ for all $s \in R$, where $[s]$ is the equivalence class of s under $\sim$. π is surjective trivially, since the only equivalence classes in Q are of form $[s]$ for some $s \in R$. To see π is monotone, observe the order on Q is $[s_1] \leq_Q [s_2]$ if and only if $s_1 \leq_R s_2$. An equivalent way to state this definition is: $\pi(s_1) \leq_Q \pi(s_2)$ if and only if $s_1 \leq_R s_2$ (this is just rewriting $[s_i]$ as $\pi(s_i)$). The "if" part proves the monotonicity of π. $\square$

Lemma 9 permits us to adapt the poset decomposition theorem (theorem 5) to preorders. If R is a preorder, form the quotient $Q = R/\sim$, which will be a partial order. Apply the embedding lemma (lemma 7) to yield an embedding $\phi : Q \to \mathbb{N}^n$ of Q in $\mathbb{N}^n$. Composing ϕ with the projection π gives a monotone function $\phi \circ \pi : R \to \mathbb{N}^n$ which we call a *pseudo-embedding* of R into $\mathbb{N}^n$. By this we mean the following. Write ψ for $\phi \circ \pi$. ψ is such that $\psi(s_1) = \psi(s_2)$ exactly when $s_1 \sim s_2$. Consequently, ψ behaves like an embedding, except that it sends equivalent individuals in R to the same point in $\mathbb{N}^n$. Otherwise, it sends non-equivalent elements in R to different points in $\mathbb{N}^n$ while preserving the order relations between them. Let us record these observations as:

Theorem 10 (Preorder Decomposition Theorem). *Every finite preorder can be pseudo-embedded into $\mathbb{N}^n$ (and thus into $\mathbb{R}^n$). Moreover, every finite preorder has a minimum n, the dimension of the preorder, for which such pseudo-embedding is possible.*

Let us examine an example to help visualize the definitions.

Example 11. Consider the following game:.

p	$rock$	$stone$	$paper$	$scissors$
$rock$	0	0	−1	1
$stone$	0	0	−1	1
$paper$	1	1	0	−1
$scissors$	−1	−1	1	0

This game is rock-paper-scissors with a clone of rock called "stone." In this case, $S = T = \{rock, stone, paper, scissors\}$, and $R = \{-1, 0, 1\}$ with $-1 \leq 0 \leq 1$.

By comparing the rows of this matrix, we can see that none of the strategies dominates any of the others. Every strategy does well against at least one opponent; likewise, every strategy does poorly against at least one opponent. However, $rock \sim stone$ because their rows are identical. Consequently, the induced preorder on $\{rock, stone, paper, scissors\}$ contains only the relations $rock \leq stone$ and $stone \leq rock$. We show this preorder in figure 1.

When we mod out the equivalence relation $\sim$, we arrive at a partial order consisting of the three equivalence classes $\{[rock], [paper], [scissors]\}$. In this partial order, all three elements are incomparable to one another. This is, in fact, a 2-dimensional partial order, with linear realizer $\mathcal{L} =$
$\{([rock], [paper], [scissors]), ([scissors], [paper], [rock])\}$ Figure 2 shows a plot of the corresponding pseudo-embedding.

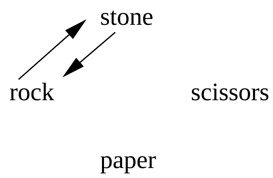

Fig. 1. The preorder rock-stone-paper-scissors displayed as a graph. An arrow between two individuals indicates a $\leq$ relationship; absence of an arrow indicates the individuals are incomparable.

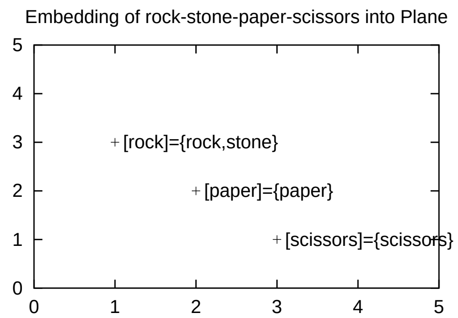

Fig. 2. The preorder rock-stone-paper-scissors pseudo-embedded into the plane $\mathbb{N}^2$

2.3 Discussion

We began with a co-evolution problem which may have had intransitive cycles and other pathologies. We ended up with what amounts to an embedding of the problem into $\mathbb{N}^n$ for some unknown dimension n. $\mathbb{N}^n$ is a particularly simple partial order; in particular, it is transitive. How did we turn a pathological problem into a nice transitive one?

The first point to note is that a pseudo-embedding, while well-defined mathematically, is not easily computable. Indeed, it is has been known for some time that even learning the dimension of a poset is NP-complete [7]. Furthermore, in order to compute a poset decomposition, we need to have all the elements of the poset on hand. While in many problems it is possible to *enumerate* all solutions,[4] it is typically intractable to do so. Thus, while pseudo-embeddings exist as mathematical objects, they are at least as expensive to compute as solving the problem by brute force. This is not surprising; if we had a pseudo-embedding in hand, we could treat our problem as a greater-than game and solve it relatively easily. Pseudoembeddings thus encode a lot of information about the original problem which we have to "pay for" somehow. In essence, what we have done is reorganize the information in a problem.

A second point is that we can at least hope that the preorder decomposition of a co-evolution problem can be *approximated* by a more practical algorithm. In the next section, we will present a simple algorithm aimed at approximating the preorder de-

[4] Exceptions include problems with real-valued parameters.

composition of a problem and show that it works reasonably well on two instances of the numbers game.

3 Experiments

In this section we present our experimental results. We begin by recalling the numbers game, from [1]. Next, we describe the algorithms we employ and our implementation choices. Finally, we present and discuss results.

3.1 The Numbers Game

"The numbers game" [1] actually refers to a class of games. Common among them is that the set of individuals, call it S, consists of n-tuples of natural numbers. How we choose to compare individuals, and what choice we make for n, define an instance of the game. In our experiments, we will be considering two instances. In both, we will deviate somewhat from the score functions defined in [1]. Instead of returning a score, we will construct our functions to have form $p : S \times S \to \{0,1\}$, where $S = \mathbb{N}^2$ is the set of ordered pairs of natural numbers, and the function p simply says which individual is bigger (i.e., gets a bigger score or "wins" the game).

In our experiments we only present data for 2-dimensional problems, since the issues we wish to emphasize are already visible at this low dimensionality. For simplicity of presentation, we define these games for 2 dimensions only.

The Intransitive Game [IG] In this game, we first decide which dimensions of the individuals are most closely matched, and then we decide which individual is better on that dimension. The payoff function we use is:

$$p_{IG}((i_1, j_1),(i_2, j_2)) = \begin{cases} 1 & \text{if } |i_1 - i_2| > |j_1 - j_2| \text{ and } j_1 > j_2 \\ 1 & \text{if } |j_1 - j_2| > |i_1 - i_2| \text{ and } i_1 > i_2 \\ 0 & \text{otherwise} \end{cases} \tag{1}$$

This game is intransitive; one cycle is $(1,6)$, $(4,5)$, $(2,4)$ [1]. $(1,6)$ and $(4,5)$ are closest on the second dimension, so $(1,6) > (4,5)$. $(4,5)$ and $(2,4)$ are closest on the second dimension also, so $(4,5) > (2,4)$. However, $(2,4)$ and $(1,6)$ are closest on the first dimension, meaning $(2,4) > (1,6)$. Nevertheless, an individual with high values on both dimensions will tend to beat more individuals than one without, and it seems, intuitively, that the best solutions to this game are such individuals.

The Focusing Game [FG] In this game, the first and second individuals are treated asymmetrically. The second individual is scanned to see on which dimension it is highest. Then, it is compared to the first individual. The first individual is better if it is higher on the best dimension of the second individual. As a payoff function:

$$p_{FG}((i_1, j_1),(i_2, j_2)) = \begin{cases} 1 & \text{if } i_2 > j_2 \text{ and } i_1 > i_2 \\ 1 & \text{if } j_2 > i_2 \text{ and } j_1 > j_2 \\ 0 & \text{otherwise} \end{cases} \tag{2}$$

Note that this game is transitive. However, the emphasis on one dimension at the expense of others encourages individuals to race on one of the two and neglect the second. Nevertheless, an individual which is high on both dimensions will beat more individuals than one which is focused on a single dimension.

This game is closely related to the COMPARE-ON-ONE game described in [2].[5]

3.2 Algorithms and Setup

We will compare two algorithms, a *population based co-evolutionary hillclimber* (P-CHC) and a *population based Pareto hillclimber* (P-PHC). The P-CHC has a single population. To assess the fitness of an individual, we sum how it does against all other individuals according to the game we are testing. The wrinkle is that each individual produces exactly one offspring, and the offspring can only replace its parent if it is strictly better. This algorithm uses a subjective fitness measure to assess individuals, but the constraint that an offspring can only replace its own parent is a simple form of diversity maintenance resembling deterministic crowding [8].

Our population based Pareto hillclimbing algorithm is similar in spirit to the DELPHI algorithm presented in [2]. Our P-PHC operates as follows. There are two populations, candidates and tests. The candidates are assessed by playing against all the tests. Rather than receiving a summed fitness, they receive an outcome vector, as in evolutionary multi-objective optimization [9]. The outcome vectors are then compared using Pareto dominance: candidate a is better than candidate b if a does at least as well as b does versus all tests, and does better against at least one. The tests are assessed differently, using an approximation of *informativeness* [10]. Since the outcome order is $0 < 1$, the informativeness measure presented in that paper collapses to simply counting how many pairs of candidates a test says are equal.[6] In other words, each test has a score $f(t) = \sum_{s_1,s_2 \in S_i} \delta(p(s_1,t), p(s_2,t))$, where S_i is the current population, and $\delta(p(s_1,t), p(s_2,t))$ returns 1 if $p(s_1,t) = p(s_2,t)$, 0 otherwise. For our experiments, p will be one of p_{IG}, or p_{FG}. As in the co-evolutionary hillclimber, in the Pareto hillclimber individuals receive only one offspring, and an offspring can only replace its parent. However, the climbing is done separately in the two populations.[7]

In order to focus more closely on domain-specific problems, we do away with a bitstring genotype. In our experiments, genotype=phenotype. Individuals are simply pairs of numbers. To create a mutant, we add random noise to each coordinate with some probability. Notice there is no mutation bias in any particular direction.

We will be using a population size of 100 for all experiments. In the co-evolutionary hillclimber, the single population will be 100 individuals; in the Pareto hillclimber, there will be 50 candidates and 50 tests. The mutation rate is 100%; mutation adds +1 or -1 to each dimension. No form of crossover is used. We ran each simulation for 500 time steps.

[5] But note that p_{FG} has range $\{0,1\}$ and requires strict inequalities for a 1 output, whereas COMPARE-ON-ONE has range $\{-1,1\}$ and does not require strict inequalities.

[6] Strictly speaking, this statement is not true; however, two tests which give different counts of equal candidate pairs are incomparable; thus we use it as a heuristic.

[7] We should remark that neither of these algorithms was intended to be practical; rather, they are intended to test our ideas.

3.3 Results

Figures 3 and 4 show performance versus evolutionary time for both P-CHC and P-PHC on the payoff functions p_{IG} and p_{FG}. Note that the co-evolutionary hillclimber out paces the Pareto hillclimber. The Pareto hillclimber must adjust not only its candidates to make an improvement, but also its tests. Updating the tests causes a time lag which slows down progress. The graphs are intended to be qualitative, however; what is important is that both algorithms make steady progress.

Note figure 3, the intransitive game. Unlike the algorithm used in [1], P-CHC made continuous progress on the intransitive game. Since P-CHC essentially adds only a diversity maintenance mechanism, it seems the diversity is important to the success or failure of co-evolution on this problem.

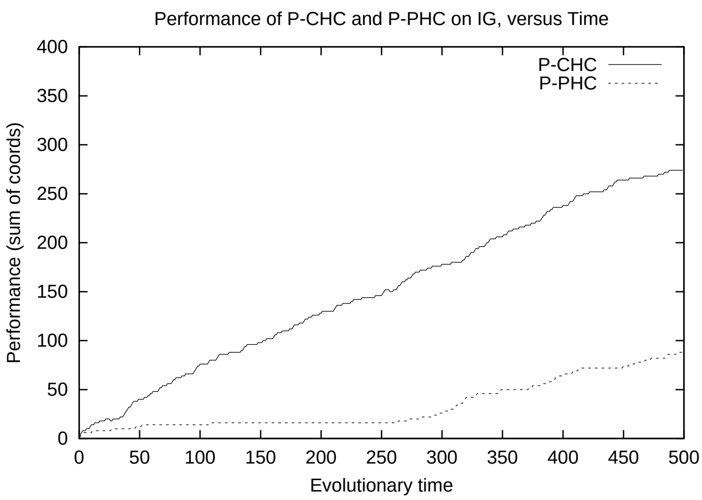

Fig. 3. Performance versus time of P-CHC and P-PHC on IG (intransitive game). Plot shows a single, typical run. Performance is measured as the sum of the coordinates; the plot shows this value for the best individual of the population at each time step.

At first glance, the Pareto co-evolution mechanism of P-PHC does not seem to be adding anything. To understand more completely what is happening, we plot in figure 5 the final candidates P-CHC found on a typical run on p_{FG}, together with the final candidates and tests which P-PHC found on a typical run. Notice how P-CHC has focused entirely on the horizontal dimension. While it made great progress there, it neglected the vertical dimension entirely. By contrast, P-PHC has maintained progress on both dimensions equally. While it did not move as far as P-CHC, it did remain balanced. Most important are the tests P-PHC found. In the plot, the tests appear to be "corralling" the candidates, keeping them in a tight group near the main diagonal. An animation of a typical run of P-PHC reveals this is indeed the case. The tests keep step behind the candidates. The same configuration of tests and candidates persists, but the group of them

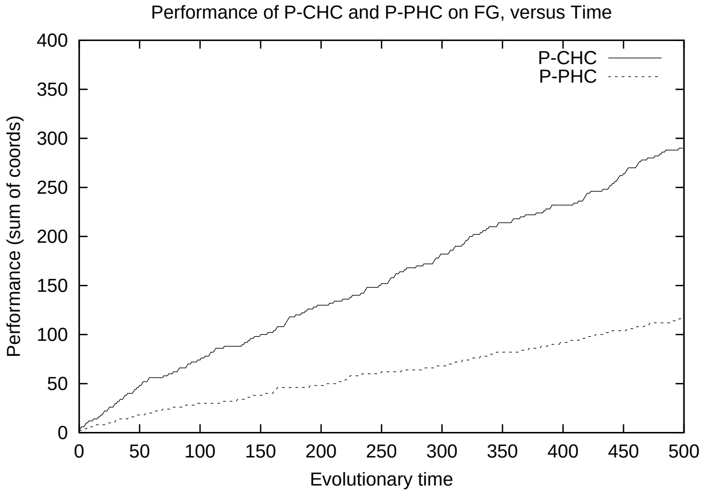

Fig. 4. Performance of P-CHC and P-PHC on FG (focusing game). Plot shows a single, typical run. Performance is measured as the sum of the coordinates of the best individual.

move slowly up and to the right, towards the better values of this game. Intuitively, we imagine the individuals found by P-CHC are brittle specialists, whereas the individuals found by P-PHC are more robust generalists.

4 Conclusion

To sum up, we have shown mathematically that a wide class of co-evolution problems, if properly construed, can be looked upon as *n*-dimensional, transitive, greater-than games. The trouble is that discovering *n* is an NP-complete problem in general, let alone the embedding which would permit us to convert our favorite problem domain to a nicely-behaved transitive domain. Nevertheless, we feel the mathematical result changes the face of intransitivity. We examined intransitivity experimentally, using insights gained from our mathematics, and saw that it may not be the demon it has been made out to be. Some algorithms fail because they are overspecializing on some dimensions of a game at the expense of others. While this observation is not new, the mathematical derivation of it sheds new light on the interpretation of co-evolution and suggests new algorithms which might overcome both intransitivity and overspecialization difficulties.

References

1. Watson, R., Pollack, J.: Coevolutionary Dynamics in a Minimal Substrate. In Spector, L., Goodman, E., Wu, A., Langdon, W., Voigt, H.M., Gen, M., Sen, S., Dorigo, M., Pezeshk, S., Garzon, M., Burke, E., eds.: Proceedings of the Genetic and Evolutionary Computation Conference, GECCO-2001, San Francisco, CA, Morgan Kaufmann Publishers (2001)

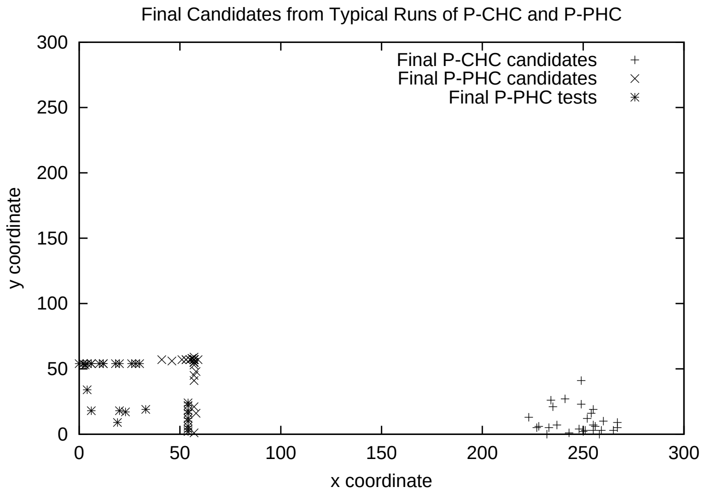

Fig. 5. Position in plane of final candidates from P-CHC run on p_{FG} (lower right), together with final candidates and tests from P-PHC (lower left). P-CHC has focused on the horizontal dimension, whereas P-PHC has improved in both dimensions. The tests which P-PHC found are arranged to "corral" the candidates along the main diagonal.

2. de Jong, E., Pollack, J.B.: Ideal Evaluation from Coevolution. Evolutionary Computation (to appear)
3. Ficici, S.G., Pollack, J.B.: Pareto Optimality in Coevolutionary Learning. In: European Conference on Artificial Life. (2001) 316–325
4. Noble, J., Watson, R.A.: Pareto coevolution: Using performance against coevolved opponents in a game as dimensions for Pareto selection. In Spector, L., Goodman, E., Wu, A., Langdon, W., Voigt, H.M., Gen, M., Sen, S., Dorigo, M., Pezeshk, S., Garzon, M., Burke, E., eds.: Proceedings of the Genetic and Evolutionary Computation Conference, GECCO-2001, San Francisco, CA, Morgan Kaufmann Publishers (2001) 493–500
5. Bucci, A., Pollack, J.B.: A Mathematical Framework for the Study of Coevolution. In: FOGA 2002: Foundations of Genetic Algorithms VII, San Francisco, CA, Morgan Kaufmann Publishers (2002)
6. Scheinerman, E.R.: Mathematics: A Discrete Introduction. 1st edn. Brooks/Cole, Pacific Grove, CA (2000)
7. Yannakakis, M.: The Complexity of the Partial Order Dimension Problem. SIAM Journal on Algebraic and Discrete Methods **3** (1982) 351–358
8. Mahfoud, S.W.: Crowding and Preselection Revisited. In Männer, R., Manderick, B., eds.: Parallel Problem Solving from Nature 2, Amsterdam, North-Holland (1992) 27–36
9. Fonseca, C.M., Fleming, P.J.: An Overview of Evolutionary Algorithms in Multiobjective Optimization. Evolutionary Computation **3** (1995) 1–16
10. Bucci, A., Pollack, J.B.: Order-Theoretic Analysis of Coevolution Problems: Coevolutionary Statics. In Barry, A.M., ed.: GECCO 2002: Proceedings of the Bird of a Feather Workshops, Genetic and Evolutionary Computation Conference, New York, AAAI (2002) 229–235