

Relationships between Internal and External Metrics in Co-evolution

Elena Popovici

George Mason University
4400 University Dr.
Fairfax, VA, 22030, USA
epopovic@gmu.edu

Kenneth De Jong

George Mason University
4400 University Dr.
Fairfax, VA, 22030, USA
kdejong@gmu.edu

Abstract- Co-evolutionary algorithms (CEAs) have been applied to optimization and machine learning problems with often mediocre results. One of the causes for the unfulfilled expectations is the discrepancy between the external problem solving goal and the internal mechanisms of the algorithm. In this paper, we investigate in a principled way the relationships between the internal subjective metric used as fitness by a co-evolutionary algorithm and the external objective metric measuring the algorithm's progress towards the envisioned goal. We point out the complexity of these relationships and explain their causes.

1 Introduction

Co-evolution emerged as a new subfield of EC at the intersection of efforts coming from several directions: 1) the simulation of those processes taking place in nature where reciprocal evolutionary change occurs between interacting species or populations; 2) the need for a computational method to tackle domains in which performance of a potential solution can only be expressed based on interactions with other potential solutions; and 3) improving EAs' performance through problem decomposition.

Co-evolutionary algorithms have thus been implemented and applied for different purposes, including simulation of real-world co-evolutionary systems, study of phenomena emerging in such systems [1] and, predominantly, problem solving ([2, 3, 4, 5], etc.). Results have varied from encouraging to disappointing across all goals.

In this paper we analyze one of the main causes of disappointment when applying co-evolution to problem solving, namely the disconnection between the external goal envisioned and the internal ways of working of co-evolution.

2 Background

2.1 Co-evolution Goals

In the past, co-evolution was often applied to various problem domains without a clear *a priori* specified goal. The *loose* goal was to see whether it could produce any *interesting* results in a particular domain. What interesting meant was decided post-factum and at times arbitrarily (e.g. humans putting smartness labels on robot behaviors).

For many domains there can be multiple and quite different goals. For example in win/lose game playing domains ([6, 7], etc.) one goal may be to find a player that can win against all other players (or at least as many as possible).

Another goal may be to beat some known *expert* players. Yet a different goal may be to beat humans. One cannot expect to use the same algorithm to achieve all of them; yet in the past it happened that the same co-evolutionary algorithm was used for different goals.

One of the reasons co-evolution was used in the first place was that the envisioned goal could not be translated into a fitness function for a traditional EA to use. This could be for one of several reasons: 1) the goal was not even theoretically testable (e.g. find a chess playing program beating all other possible programs); 2) testing the goal was computationally intractable (e.g. find a sorting network that sorts all binary input sequences, when the number of inputs is high); 3) the goal was computationally testable, but it was expensive to do so for each evaluation during the algorithm (e.g. in the sorting networks domain when the number of inputs is small [4]); and 4) the goal may not translate into an evolution-friendly fitness function (e.g. evaluating against a set of expert opponents may provide no gradient for the search). The hope of co-evolution was to have an algorithm achieve such a goal without specifically encoding it internally into fitness. Was this hope realistic?

On one hand, there seem to be no strong reasons to believe that an algorithm playing computer programs against computer programs would evolve human-competitive players, other than the fact that human players can be seen as the results of an evolutionary process as well. Sometimes co-evolution gave good results in this respect and sometimes *it didn't*. On the other hand, there seem to be stronger connections between an algorithm internally playing computer programs in a limited number of games (against other evolving computer programs) and the goal of optimality over all possible games. It turned out that co-evolution may or *may not* work well in this respect either. The relationships between the internal fitness (metric) used by the algorithm and its progress towards the goal (measured by some external metric) are generally not clear.

In this paper we analyze such relationships between internal and external metrics in co-evolution. We do so for a basic co-evolutionary algorithm and domains with a computationally testable goal (i.e. computable external metric). We also provide insight into the causes for the observed relationships by means of dynamical systems analysis. We believe we need to understand such simple setups first before moving on to more complex (e.g. untestable) goals or more complex algorithms. In fact, as the reader will see, even in basic setups there is a wide range of possible relationships.

2.2 Co-evolution Metrics

It is important to clearly define goals before applying co-evolution and also to define ways of instrumenting it. A lot of confusion persisted in the field due to failure to distinguish between the internal and the external perspective. [5] cleared matters by providing a clustering of metrics for co-evolution. We reproduce it here in a slightly different form.

Instrumenting co-evolution comes down to defining some metrics of the type $m(\textit{individual}, \textit{context})$ and analyzing the values produced by these metrics for some/all individuals produced at run-time by the algorithm. There are two important criteria for clustering such metrics: dependence on context and influence on the algorithm.

Based on the first criteria, metrics can be divided into *objective* and *subjective*. An objective metric will always return the same value for an individual, regardless of the context. A subjective metric is one that is not objective.

Based on the second criteria, metrics can be divided into *internal* and *external*. An internal metric is one whose values are used by the algorithm and influence its course. An external metric is one that is not internal.

A traditional EA uses an internal objective metric as fitness. A co-evolutionary algorithm always uses an internal subjective one. The context is usually a set of other evolving individuals. External metrics can also be objective or subjective. In cases where co-evolution was applied without a precise goal or with a non-testable goal, some external subjective metrics of performance were used to tell whether the algorithm was making any *progress* (e.g. metrics that evaluate individuals in contexts composed of individuals from previous generations of the same run - [7], etc.). While such metrics tell us something about the behavior of the algorithm, they may not tell us how the algorithm is doing with respect to the actual goal. Performance with respect to a goal should be measured with an external objective metric.

Of the variety of goals for which co-evolution was applied, we choose to investigate one which is characteristic of problem solving domains involving two different species, such as evolving sorting networks vs. input sequences [4] or cellular automata vs. initial conditions [3]. The goal is to find the individual of the first species with the highest cumulative payoff over interactions with all individuals of the other species (e.g. networks that sort all binary input sequences; CAs that correctly classify the maximum number of initial conditions).

First, we will define a proper external objective metric to measure performance with respect to this goal. Then we will analyze the relationships between the run-time trends of this external objective metric and the algorithm's internal subjective metric, as well as their causes.

The only previous study in this area is [6], which analyzed relationships between internal and external metrics in the context of some number game domains. However, this research cannot be directly extended to other domains, as the external objective metrics are problem dependent. Also, it did not provide explanations for the observed behavior. We address both these issues in this paper.

3 The Setup

3.1 The Domains and the External Objective Metric

In the experiments reported in this paper we focused on competitive co-evolution involving two species.¹ We call the two species the X -specie and the Y -specie and both contain real numbers. Each problem domain is expressed by a two-parameter function $f : D_X \times D_Y \rightarrow R; D_X, D_Y \subset R$. When an x -individual interacts with a y -individual, the outcome is that the x -individual receives a payoff equal to $f(x, y)$ and the y -individual receives a payoff equal to $\max_{(x,y) \in D_X \times D_Y} f(x, y) - f(x, y)$. Outcomes of interactions between individuals can thus be characterized in terms of game theory as *constant sum*.

Both species want to maximize payoff, however our problem solving goal is only related to the X species. Namely, the *goal* is to find an x -individual with maximum *expected* (or *average*, or *cumulative*) payoff over all possible interactions with y -individuals. Formally, this means finding x_0 which maximizes $e(x) = \int_{D_Y} f(x, y) dy$. $e(x)$ represents the cumulative payoff for an x -individual. The average (or expected) payoff is given by $avg(x) = e(x)/size(D_Y)$. $avg(x)$ will be our *external objective metric* of performance with respect to reaching the goal.

We use for our experiments four domains described by four such functions, three of which were previously introduced in [8], where their analytical forms can be found (for space constraints we do not reproduce them here). A fourth one, *chaoticRidges*, is introduced here in order to expose additional possible relationships between metrics.

Its expression is: $chaoticRidges_n(x, y) = \begin{cases} n + 2|\frac{2n}{3} - y| - |x - y| & \text{if } (x - \frac{2n}{3})(y - \frac{2n}{3}) \geq 0; \\ \frac{n+x}{3} + |2\frac{n+x}{3} - y| & \text{if } x \leq \frac{n}{2}; \\ -n + 3x + |2(n-x) - y| & \text{if } x \leq \frac{2n}{3}; \\ 3(n-x) + |2(n-x) - y| & \text{otherwise.} \end{cases}$

In all the experiments presented here $n = 8$ was used (i.e. $D_X = D_Y = [0, 8]$). Three dimensional views of all four functions are shown in Figures 1 and 2. Each surface in these figures represents the payoff that x -individuals get from interacting with a y -individual for that particular setup. The payoff surface for y -individuals is omitted for brevity and it would look like the X payoff surface upside down.

Figures 1 and 2 show what the landscapes look like from the point of view of individual interactions (and therefore internal fitness). They are very much alike with regard to ruggedness/modality. By contrast, Figures 3 and 4 show us what the landscapes look like from the point of view of our external objective metric, the one we really hope to optimize. They plot for each of the four functions $avg(x)$ vs x . These plots confirm that the domains are similar. They all have a single global maximum, no local maxima and a smooth gradient. We shall see however that relationships between the external objective metric and the internal subjective one (for best of generation individuals) can vary widely across these domains.

¹We expect the same richness of behaviors for more than two species/populations, but analysis/visualization are more difficult.

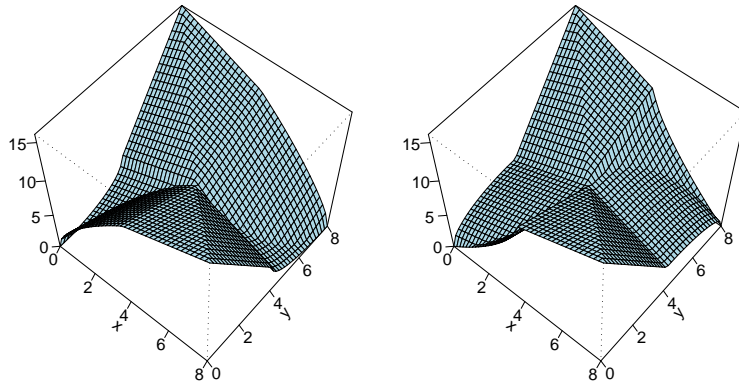


Figure 1: Internal landscape perspective (individual interaction payoff surfaces); left: *collapsingRidges*, right: *expandingRidges*

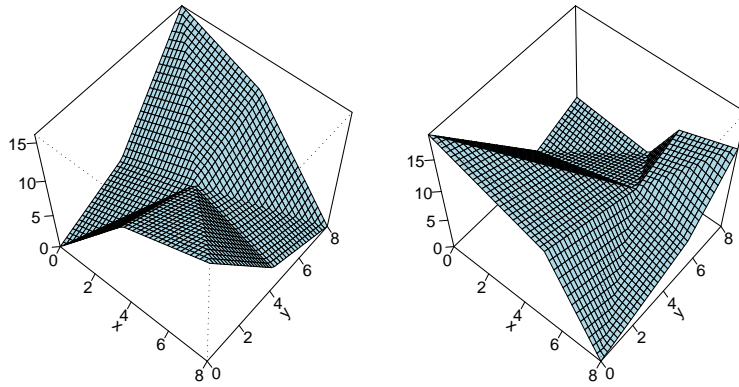


Figure 2: Internal landscape perspective (individual interaction payoff surfaces); left: *cyclingRidges*, right: *chaoticRidges*

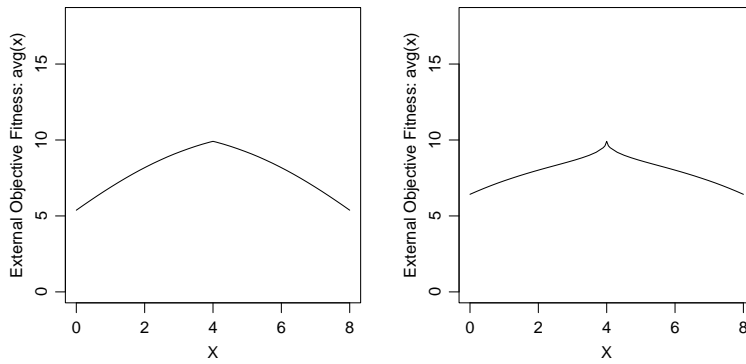


Figure 3: External landscape perspective (average payoff curves); left: *collapsingRidges*, right: *expandingRidges*

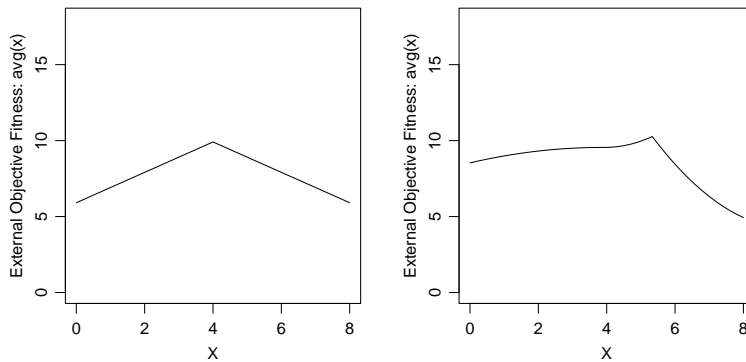


Figure 4: External landscape perspective (average payoff curves); left: *cyclingRidges*, right: *chaoticRidges*

3.2 The Algorithm and the Internal Subjective Metric

The co-evolutionary algorithm used in these experiments involves two populations, one for each species and it takes turns in evolving them, i.e. only one population is active per generation. Within each population a non-overlapping generational model is used, with elitism of 1. The individuals in the active population are evaluated by having them interact with the best individual in the other population at the previous generation (*last elite opponent* evaluation [9]). This defines our *internal subjective metric*. Binary tournament selection is followed by Gaussian mutation with sigma fixed to 0.25, altering each individual with probability 0.75. The new individuals are evaluated in combination with the best individual in the other population and, based on the fitnesses obtained, the best individual for this population is determined. Control is passed to the opposite population.

Unless otherwise specified, each population had a size of 100. At the beginning of the algorithm both populations are initialized uniformly random across the domain. The X -population is the first to be active. One random y -individual is picked for evaluating all x -individuals, as there is no *best* in the Y population yet. The first Y -generation (second of the run) uses for evaluation the actual best x -individual determined in the previous (X) generation.

4 The Results

Since we are set in a context of applying co-evolution for optimization, we will instrument our algorithm from the perspective of change in best individuals throughout the run.

For each experiment we conducted 100 runs and generated plots for all of them. These plots were then visually inspected. If all had the same trend, we picked to show in the paper the one that displays the trend most clearly. If there were several different trends, we similarly picked one from each category, regardless of which category was more heavily represented. Mean fitness trends (not shown) were either presenting nearly no variance (for *collapsingRidges* and *expandingRidges*) or they were obscuring the true phenomena, because they were averaging very different values (for *cyclicRidges* and *chaoticRidges*).

We grouped the results into three categories, based on the relationship between the internal and the external metric. In comparing the metrics, we looked both at the relationship between their respective values and at the nature of their change over time. For each case, we show how the dynamics analysis explains the (otherwise mysterious) trends.

4.1 Opposite Monotonic Trends

The Fitness Figure 5 shows a typical run for the *collapsingRidges* function. On the left hand side, the fitness curves plot (only) the points corresponding to the best individual in generations in which X was the active population. For each such point, the internal subjective metric used by the algorithm is the payoff that the best x -individual in that generation obtained from interacting with the best y -individual from the previous (non-plotted) generation. The external

objective metric is $avg(x)$. Note that as the external metric is an average over the domain, its range of values is more restricted than that of the internal metric.

In this case the internal subjective metric decreases from its initial value quite fast in the first few generations and then stabilizes for the rest of the run at a value in the middle of its range. This might suggest very poor performance, but in fact the external metric has the exactly opposite trend, i.e. it increases quickly at the beginning and then stabilizes at what is actually its maximum value.

A completely reversed situation happens for the *expandingRidges* function. As can be seen in Figure 6, initial generations display rapid increase in the internal metric and rapid decrease in the external metric, after which they both stabilize (the internal one at its maximum and the external one at its minimum). What internally looks like progress, from an external perspective is actually regress.

At this point the situation appears confusing. The two functions are very similar from several perspectives. First, the 2D surfaces generated by the payoff of individual interactions have the same degree of ruggedness (same global optima, no local optima, same number of ridges, smooth gradient). Second, the curves generated by the average payoff over all interactions have the same degree of ruggedness (single global optima, no local optima, smooth gradient). Why does the same algorithm have opposite behaviors?

The Dynamics To elucidate this, we look at the dynamics of best individuals from the perspective of their movement across the search space, using a technique introduced in [8] which we will briefly summarize here.

The right hand side of Figures 5 and 6 shows the trajectories of best individuals across the space (for the same runs). Each inflection point on such a trajectory corresponds to one generation and it displays the pair (of genotypes) of the best individual from that generation and the best individual from the previous generation (from interaction with whom it got its fitness). The starting point is marked by an empty geometrical shape (in this case a circle) and the end point by the same shape but filled.

The plots also display the so called *best response curves* [8] characteristic to the surfaces of individual interactions payoff and how these curves influence the trajectory of the best individuals. The *bestResponseX* curve is obtained by plotting for every y the x that gets the highest payoff out of interacting with that particular y . The *bestResponseY* curve is similarly defined.

To understand the fitness plot for a function, one needs to collectively analyze the trajectory plot, the individual interactions surface plot and the external metric plot for that particular function. For any function, a point of intersection of the two best response curves is a fixed point for best individual trajectories. In the case of the *collapsingRidges* function, due to the fact that in this point the absolute value of the slope of the *bestResponseY* curve is smaller than that of the *bestResponseX* curve ($|0| < |-1|$), this fixed point has attracting behavior. Regardless of where a trajectory starts (we show one starting quite far away from this point),

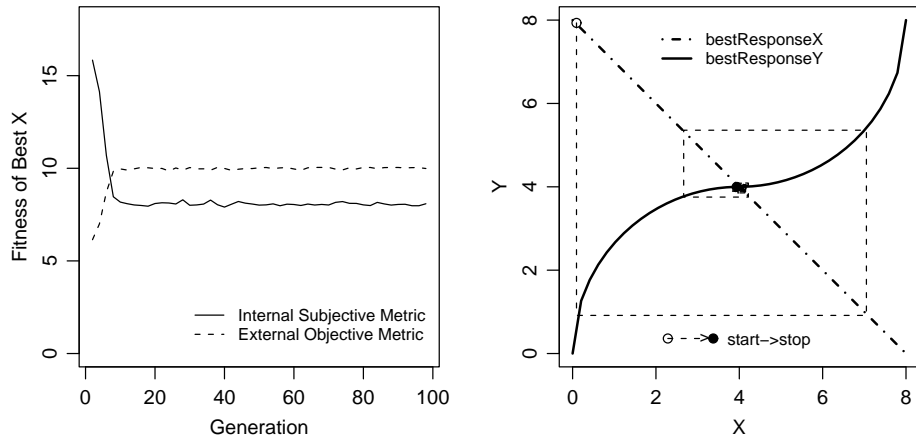


Figure 5: A selected run for *collapsingRidges*; left: best-of-generation metrics, right: best-of-generation space dynamics.

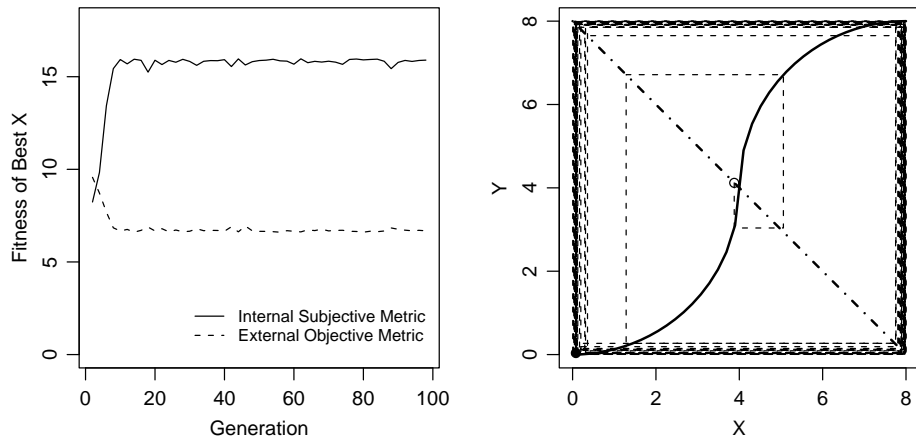


Figure 6: A selected run for *expandingRidges*; left: best-of-generation metrics, right: best-of-generation space dynamics. (On the right side, the legend was omitted to avoid clutter, but it is the same as on the *collapsingRidges* plot above.)

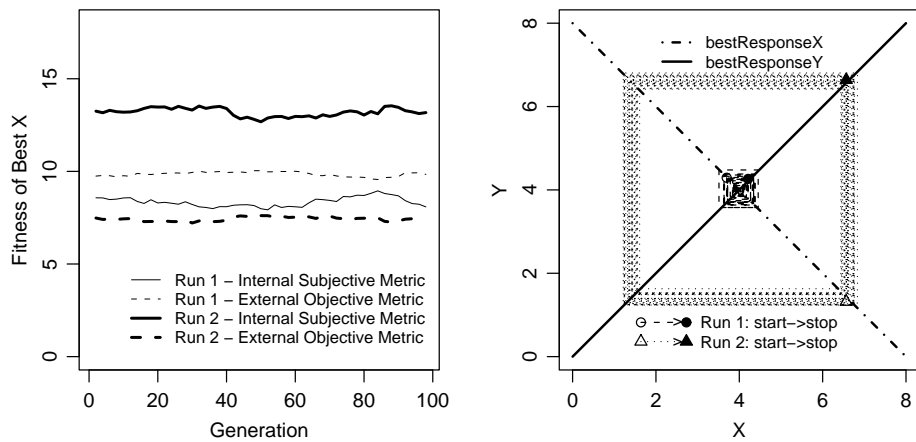


Figure 7: Two selected runs for *cyclicRidges*; left: best-of-generation metrics, right: best-of-generation space dynamics.

it will end up in the fixed point. The fitness plot basically reports internal and external metric values for the points of the trajectory on or near the *bestResponseX* curve. These points move closer and closer to the center of the space. The internal fitness values correspond to the values associated with these points on the 2D surface of individual interactions payoff (left in Figure 1). We can see on this surface that moving closer to the center means decreasing values. Once at the center there is only small variation due to stochastic effects. The external fitness values correspond to the values associated with the X coordinate of these points on the external fitness chart (left in Figure 3). We can see on this curve that moving from the sides to the center generates increase in values up to the maximum in the fixed point (again, with small perturbations due to stochastic effects). And this is exactly what we see on the fitness plot.

In the case of the *expandingRidges* function, the relationship between the absolute values of the slopes of the best response curves at the fixed point is reversed ($|\infty| > |-1|$). This makes the fixed point unstable, so trajectories are repelled from it (even when they start very close to it, as in the example run presented). Following the same type of analysis as for the *collapsingRidges*, we now understand the shapes of the internal and external fitness curves and why they are different from the previous function (both the 2D surfaces and the external metric curves have the same monotonicities, but the trajectories move in one case from the center to the extremes and vice versa in the other case).

4.2 Flat Trends

On the *cyclingRidges* function the two metrics seem to agree in their trends, in the sense that both of them stagnate from beginning to end. However, they disagree in the type of values they report. Figure 7 shows two runs superimposed on the same plot(s) (runs are distinguished by line type/width). In one of them the internal metric reports high values for its range while the external metric reports low values for its own range. In the other run, the internal metric shows average values, while the external one is almost at its optimum.

Armed with the technique described above, we can explain these results. The dynamics plot shows that the two best response curves have the same slope in absolute value. This causes the fixed point to be neither attracting nor repelling. Additionally, it causes the appearance of an infinite number of size two periodic orbits. Trajectories wander about such orbits close to the point where they started (they do not follow a single orbit due to stochastic effects). The periodic orbits intersect the *bestResponseX* curve in points symmetric with respect to the center, and the corresponding fitness values are equal both for the 2D surface of individual payoffs (internal metric) and for the average payoff external metric. This explains the flat trend of the fitness curves.

Whether the internal metric claims better performance than the external or vice versa depends on the point where a particular run's trajectory starts and is cycling close to. Trajectories wandering close to the center will have an internal metric close to 8 (the value at the center) which is in the middle of the 0-16 range and an external metric close to

10, which is the maximum in a range of 6-10. Trajectories wandering closer to the extremes of the space will have the internal metric closer to its maximum of 16 and the external metric closer to its minimum of 6.

4.3 Complex Trends

In the three examples examined so far, the trend of each metric throughout the run was monotonic, apart from minor oscillations. The fourth function paints quite a different picture, as can be seen in Figure 8 which displays three different runs, each with its own set of plots. Both the internal and the external fitness curves are very rugged, with numerous spikes. For the first run, the spikes are irregular in shape, size and frequency throughout the run. The second run displays two phases, first one of irregular spikes and then one of very regular spikes. During this second phase, the two metrics go up and down synchronously, whereas in the first phase (and more clearly visible in the plot of the third run) when external fitness increases, internal fitness decreases and vice-versa. Finally, the third run displays fairly regular spikes and the two metrics are negatively correlated.

Next, we investigate the dynamics and tie them to the fitness. The right hand side plots in Figure 8 show clearly more complex behavior than previously seen. Considering the system $x \rightarrow \text{bestResponseX}(\text{bestResponseY}(x))$, that deterministically follows the best response curves, the following can be mathematically proven: 1) it has a repelling fixed point at $(\frac{16}{3}, \frac{16}{3})$; 2) it has periodic orbits of any size; 3) all periodic orbits are sources (i.e. repelling) 4) it has chaotic orbits; and 5) the whole interval $[0, 8]$ is a chaotic attractor. The proof is beyond the scope of this paper, but for the interested reader it is similar to those for the logistic and tent maps presented in chapters 1 and 3 of [10].

Of course, when we introduce stochasticity, as in our CEA, we no longer have clear definitions of the above mentioned dynamical systems concepts. However, visually at least, the CEA using a population size of 100 approximates quite closely the deterministic system.

The first run is the analog of a chaotic orbit for the corresponding deterministic system (which would come infinitesimally close to any point in the $[0, 8]$ interval an infinite number of times, without periodicity). This explains the irregular nature of the fitness curves for this run. The first part of the second run is fairly similar to it, then the trajectory spends some time around a period two orbit, generating the saw-like shape of the fitness. In the third run the trajectory wanders around a higher period orbit, which explains the wider regularly shaped teeth in the fitness plot.

Note first that values in the fitness plot come from points along the *bestResponseX* curve (the main diagonal of the space). Along this diagonal, the minimum payoff is 8 (see Figure 2), therefore the internal fitness never goes below this value. At the same time, the maximum average payoff is 10 (see Figure 4). Points close to the left/bottom end of the diagonal (x close to 0) have close to maximum internal fitness and high but not quite optimum external fitness. Points near the fixed point have very low internal fitness and close to optimum external fitness. Points close to the right/top end

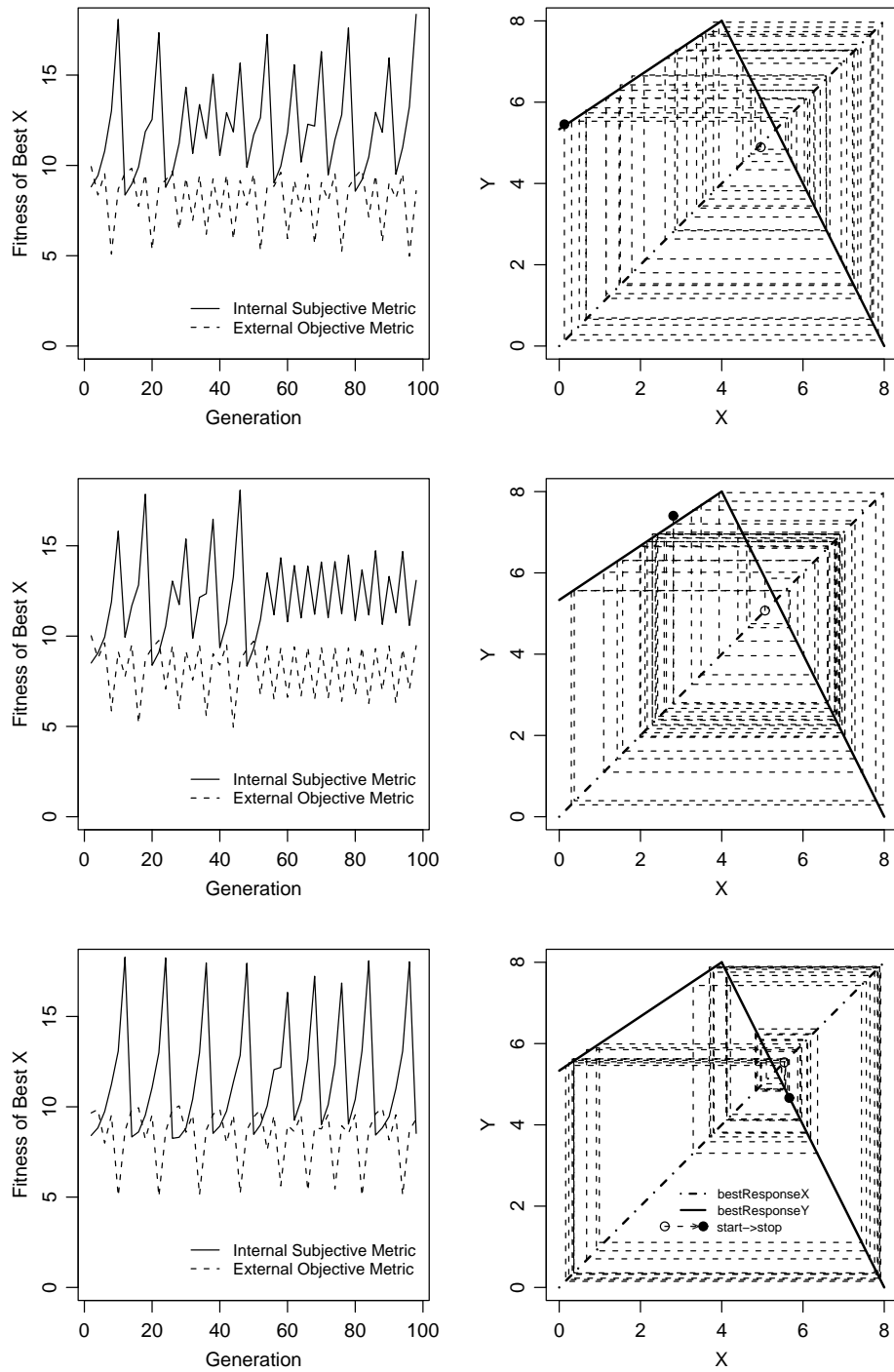


Figure 8: Three qualitatively different runs for *chaoticRidges*; left: best-of-generation metrics, right: best-of-generation space dynamics. (To avoid clutter, the legend for the dynamics plots is shown only for the third run - bottom right.)

of the diagonal have medium internal fitness and very low external fitness. The rest is somewhere in between.

As can be seen on any of the trajectory plots, $bestResponseX(bestResponseY(x))$ has the following general effects: 1) it moves points near the top/right end to points near the bottom/left end; 2) moves points close to the bottom/left end to points near the fixed point, but above it; 3) moves points near the fixed point, but above it to points near the fixed point, but below it; and 4) moves points near the fixed point, but below it to points further away from the fixed point and closer to the top/right end.

Consulting Figures 2 and 4, we can see that: case 1) corresponds to some increase in the internal metric and considerable increase in the external one (with respect to their own ranges); 2) entails deep drop in the internal metric and small increase in the external one; 3) gives small increase in the internal metric (gradient to the left of the fixed point is bigger than gradient to the right) and almost no change in the external one; and 4) corresponds to some increase in the internal metric and some decrease in the external one.

This causes a wide range of metric relationships. First of all, the metrics are no longer monotonic throughout the whole run, although they can be for smaller periods of time (e.g. the upward slopes in internal fitness on the spikes for the third run). Second, the two metrics may no longer agree or disagree throughout the whole run, instead there can be periods when they are positively correlated (e.g. second half of second run) and periods when they are negatively correlated (e.g. the teeth in the third run). Moreover, correlation can change sign very often and irregularly (e.g. first run).

5 Conclusions, Preliminary and Future Work

In this paper we used four simple fitness landscapes to gain insight into one of the main causes of failure when using co-evolution for optimization, namely the disconnection between the envisioned external goal and the internal behavior of the algorithm. We identified a problem solving goal that appears often in practice and formalized it in a domain independent way. Moreover, the goal is expressed in terms of interactions between individuals, as these are the ones used by the algorithm. This way, there is at least a common ground between the goal and the algorithm. We defined a computable external objective metric for testing progress towards the goal. We used a basic algorithm so that the internal subjective fitness metric had a simple dependency on the individual interactions.

We showed that even with a simple algorithm on simple landscapes, complex relationships between the two metrics can emerge. To understand them, a single perspective analysis is not enough. One must combine several views, an important one being given by dynamical system analysis.

The research conducted is preliminary in nature and we deliberately picked a simple setup. Natural further extensions include: 1) analyzing more complex evaluation schemes; 2) analyzing more complex external goals; and 3) analyzing cooperative setups.

We have already conducted some experiments for cooperative co-evolution using the same external metric. This

metric would be appropriate when the goal is not finding the ideal combination, but finding individuals that generally collaborate well with others. Previously, [5] suggested that CCEAs tend to move towards points of maximum average performance. By performing the same analysis as in this paper for the two functions introduced in [11] we can say that this is not always the case.

Bibliography

- [1] R. Axelrod, "Evolving new strategies," in *Genetic Algorithms and Simulated Annealing*, L. Davis, Ed., London, 1987-89, pp. 32-41.
- [2] D. Cliff and G. F. Miller, "Co-evolution of pursuit and evasion II: Simulation methods and results," in *From animals to animats 4*, P. Maes et al., Eds. Cambridge, MA: MIT Press, 1996, pp. 506-515.
- [3] H. Juillé and J. B. Pollack, "Coevolving the ideal trainer: Application to the discovery of cellular automata rules," in *Third Annual Genetic Programming Conference*, J. R. Koza et al., Eds. Madison, Wisconsin, USA: Morgan Kaufmann, 1998, pp. 519-527.
- [4] W. D. Hillis, "Co-evolving parasites improve simulated evolution as an optimization procedure," in *Ninth annual international conference of the Center for Non-linear Studies on Self-organizing, Collective, and Cooperative Phenomena in Natural and Artificial Computing Networks on Emergent computation*. North-Holland Publishing Co., 1990, pp. 228-234.
- [5] R. P. Wiegand, "An analysis of cooperative coevolutionary algorithms," Ph.D. dissertation, George Mason University, Fairfax, VA, 2004.
- [6] R. A. Watson and J. B. Pollack, "Coevolutionary dynamics in a minimal substrate," in *Genetic and Evolutionary Computation Conference*, L. Spector et al., Eds. San Francisco, California, USA: Morgan Kaufmann, 2001, pp. 702-709.
- [7] C. D. Rosin and R. K. Belew, "New methods for competitive coevolution," *Evolutionary Computation*, vol. 5, no. 1, pp. 1-29, 1997.
- [8] E. Popovici and K. De Jong, "Understanding competitive co-evolutionary dynamics via fitness landscapes," in *AAAI Fall Symposium on Artificial Multiagent Learning*, S. Luke, Ed. AAAI Press, 2004.
- [9] K. Sims, "Evolving 3D morphology and behaviour by competition," in *ALife IV*, R. Brooks and P. Maes, Eds. Cambridge, MA, USA: MIT Press, 1994, pp. 28-39.
- [10] T. S. K. Alligood and J. Yorke, *Chaos: An Introduction to Dynamical Systems*. Springer, 1996.
- [11] E. Popovici and K. De Jong, "Understanding cooperative co-evolutionary dynamics via simple fitness landscapes," in *Genetic and Evolutionary Computation Conference*, 2005, to appear.