Objective Fitness Correlation

Evaluating coevolutionary evaluation

Edwin D. de Jong
Institute of Information and Computing Sciences
Utrecht University
PO Box 80.089
3508 TB Utrecht
The Netherlands
dejong@cs.uu.nl

ABSTRACT

This paper introduces the Objective Fitness Correlation, a new tool to analyze the evaluation accuracy of coevolutionary algorithms. Accurate evaluation is an essential ingredient in creating adequate coevolutionary dynamics. Based on the notion of a solution concept, a new definition for objective fitness in coevolution is provided. The correlation between the objective fitness and the subjective fitness used in a coevolutionary algorithm yields the Objective Fitness Correlation. The OFC measure is applied to three coevolutionary evaluation methods. It is found that the Objective Fitness Correlation varies substantially over time. Moreover, a high OFC is found to correspond to periods where the algorithm is able to increase the objective quality of individuals. This is evidence of the utility of OFC as a measure to evaluate and compare coevolutionary evaluation mechanisms. The Objective Fitness Correlation (OFC) provides a precise analytical tool to measure the accuracy of evaluation in coevolutionary algorithms.

Categories and Subject Descriptors

F.0 [**General**]

General Terms

Algorithms, Experimentation, Performance

Keywords

Coevolution, objective fitness, subjective fitness, objective fitness correlation, OFC

1. INTRODUCTION

In this paper, a new tool for analyzing the accuracy of evaluation in coevolutionary algorithms is introduced.

Within evolutionary computation, coevolution is of interest in that it provides a potential to evaluate individuals using a limited, adaptive set of interaction partners. Interaction partners will be called *tests* here. In problems where the quality of individuals is determined by the outcome of tests, i.e. test-based problems [1], applying a standard genetic algorithm would require either testing individuals against all possible tests, which is typically infeasible; testing against a fixed set of tests, which biases the search towards a particular set of tests; or testing against a random sample of tests, which may not include the high-quality tests required to evaluate high-quality solutions. By letting the test set adapt over time to the evolving set of candidate solutions, coevolution may in principle provide limited size test sets that provide adequate evaluation for the evolving set of candidate solutions at each point in time.

While coevolution may in principle provide efficient and accurate evaluation, it is rather easy to devise coevolutionary setups that do not achieve this goal. In particular, the dynamic evaluation that is provided by adaptive coevolutionary test sets can lead to a diverse set of pathologies including disengagement, over-specialization, and cycling; see e.g. [2].

A recent insight in coevolution research is that the design of a coevolutionary setup should begin with a consideration of the desired *solution concept* [3]. A solution concept specifies which elements of the search space qualify as solutions and which do not. Examples of solution concepts include: maximizing the sum of outcomes against all possible opponents, the Pareto-optimal set, and Nash-equilibria.

Given a choice of the desired solution concept, a next question is how a coevolutionary algorithm may be set up such that it is likely to converge towards a solution. Currently, for each of the coevolutionary solution concepts that have so far been described in the literature, archive methods exist that provide a guarantee of monotonic progress given a generator of individuals. By coupling such an archive to a coevolutionary algorithm that provides new candidate solutions, or even to a random generator of candidate solutions, it can be guaranteed that a correct solution will eventually be found, given sufficient exploration.

While such theoretical guarantees are important in providing examples of robust coevolution algorithms that can overcome coevolutionary pathologies, the practical value of such guarantees remains limited as long as bounds on the computational expenses required to reach a solution are unavailable. For the practical purpose of developing *efficient* robust coevolutionary algorithms, an essential open problem

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. To copy otherwise, to republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee.
GECCO'07, July 7–11, 2007, London, England, United Kingdom.
Copyright 2007 ACM 978-1-59593-697-4/07/0007 ...$5.00.

now is how the dynamics of a coevolutionary algorithm can be set up such that the generated individuals improve the algorithm's approximation of the solution concept.

An essential ingredient in creating adequate coevolutionary dynamics is accurate *evaluation*; if an accurate estimate of the absolute, objective quality of individuals would be available, then the coevolutionary algorithm would become equivalent to a standard genetic algorithm, and be able to guarantee monotonic progress simply by performing elitist selection; see also [4].

An **objective fitness** measure is defined here as a static function that accepts an approximation to the solution concept, and returns a value that is maximal if and only if the approximation is a member of the solution concept. In addition to this formal requirement, it is desirable that the objective fitness measure express the degree to which the approximation approaches the solution concept. Clearly, the availability of an objective fitness measure would greatly simplify any coevolutionary search problem; it would reduce the coevolutionary problem to a standard genetic algorithm problem.

In test-based problems of practical interest, objective measures of quality are unavailable, as evaluating a candidate solution against all possible tests is typically computationally infeasible. There are two classes of problems however for which objective measures of quality *can* be obtained:

- Test problems for which the set of all possible tests is small. For board and other games, such as Nim for example, small variants can often be defined for which the set of all possible opponent strategies is sufficiently small to evaluate candidate solutions against all tests.
- For abstract test problems such as certain Numbers games [2], the objective quality of individuals can be derived analytically. If this can be done, an objective quality measure can be provided without performing tests.

We propose to analyze algorithms for coevolutionary evaluation by comparing the *subjective* coevolutionary evaluation they provide with the *objective* evaluation that is available in certain test problems. The **Objective Fitness Correlation** is defined as the correlation between the subjective fitness values calculated by a coevolutionary algorithm and the objective fitness that can be calculated for certain test problems. By measuring the Objective Fitness Correlation (OFC), an accurate analytical tool is obtained to evaluate and compare the accuracy of different coevolutionary evaluation mechanisms.

In this paper, the Objective Fitness Correlation is introduced and described. As a demonstration of the proposed approach for evaluating coevolutionary evaluation methods, the measure is applied to three of the evaluation methods used in coevolution work: the average outcome against current opponents; an informative method based on distinctions; and an archive-based approach. It is found that the Objective Fitness Correlation varies substantially over time. Moreover, a high OFC is found to correspond to periods where the algorithm is able to increase the objective quality of individuals. This is evidence of the utility of OFC as a measure to evaluate and compare coevolutionary evaluation mechanisms.

The remainder of the paper is structured as follows. After the introduction, related work is discussed. Next, Section 3 describes the main solution concepts used in current coevolution research. Section 4 introduces the Objective Fitness Correlation. Section 5 describes the LINT problem. The following section describes the algorithms. Section 7 reports the experiments, Section 8 describes the results, Section 9 provides a discussion, and Section 10 concludes.

2. RELATED WORK

The accuracy of evaluation in coevolution is a longstanding theme; see e.g. [5, 6, 1, 7]. A strong connection exists with the Numbers Game work by Richard Watson [2]; this work shares with it not only the structure of the test problem, but more importantly the aim to focus on and isolate certain aspects of coevolution while ruling out the influence of others.

The most closely related work of which we are aware is by Popovici and De Jong [8]. There, the internal and external landscapes of individuals in a coevolutionary setup are analyzed. These notions provide a valuable contribution to the understanding of evaluation in coevolution. A crucial property of coevolutionary algorithms however is that evaluation typically depends on a whole population of individuals rather than a single one. The approach that will be proposed here takes into account the influence of entire populations, by considering the subjective fitness resulting from evaluation against a population.

As noted in [8], with the exception of [2] surprisingly little earlier work has explored these notions, though several researchers have studied measuring progress in coevolutionary algorithms, e.g. [9, 10, 11]. A possible explanation is that the notion that objective evaluation measures exist for coevolutionary algorithms has only recently been clarified substantially, as a result of among others the development of Pareto-coevolution [12, 13] and the idea of solution concepts [3]. It is furthermore used in [14], which employs a population-based fitness approximation to monitor progress.

3. SOLUTION CONCEPTS

A central question in any coevolutionary setup is what the desired *solution concept* is: what is the intended goal of applying the coevolution algorithm, or equivalently, which elements of the search space count as solutions? The importance of this question has only recently been pointed out clearly [3].

A solution concept is a set that specifies the elements of the search space that qualify as solutions. A solution concept is algorithm-independent, and does not change over time; it specifies a static division of the search space into two subsets: solutions and non-solutions.

Some main solution concepts that have been used so far in coevolution are:

- S0: Simultaneous Maximization of All Outcomes. The first concept concept requires an optimal solution C to maximize the outcome over all possible tests simultaneously. This solution concept has a limited application scope, as for many problems there does not exist a single solution that simultaneously maximizes the outcome of all possible tests.
- S1: Maximization of Expected Utility. The second solution concept, which will be employed in this paper, is Maximization of Expected Utility (MEU). The MEU

solution concept specifies as solutions all individuals that maximize the expected score against a randomly selected opponent. This intuitive criterion is widely used, and is appropriate for many problems. It will be assumed here that every possible test is encountered with the same probability, although this can easily be generalized to non-uniform distributions. The MEU solution concept thus is equivalent to maximization of the average outcome of individuals against all possible opponents or tests. The MaxSolve algorithm [15] guarantees monotonic progress for this solution concept; when a generator of individuals such as a coevolutionary algorithm is coupled to the archive method, the archive under given conditions monotonically approaches the solution concept.

- S2: Nash Equilibrium. Game theory provides the solution concept of the Nash equilibrium. A Nash equilibrium specifies a strategy for each player such that no player can profitably deviate given the strategies of the other players. Individuals are viewed as strategies. In the mixed-strategy Nash equilibrium, strategies do not represent single individuals but probability distributions over the space of individuals.

- S3: Pareto-Optimal Set. Pareto-Coevolution [12, 13] views every possible test as an objective in the sense of Evolutionary Multi-Objective Optimization (EMOO). A candidate solution is said to *dominate*, another candidate solution if its outcomes against the tests are all at least as high, and its outcome on at least one test is strictly higher. The set of all individuals that are non-dominated trade off the different capabilities of candidate solutions in different ways.

- S4: Pareto-Optimal Equivalence Set. The Pareto Optimal set may contain many equivalent candidates that each solve the same combination of tests, where *solving* means obtaining a positive outcome. To address this redundancy, the Pareto-Optimal Equivalence Set is defined by the requirement that for each combination of tests that can be solved, it contains at least one candidate solution that solves it. Since multiple such sets may exist, solution concept $S4$ is defined as the collection of all such sets.

4. OBJECTIVE FITNESS CORRELATION

In this section the Objective Fitness Correlation (OFC) is defined. To this end, we first define the notions of objective and subjective fitness on which it is based.

4.1 Objective fitness measures

An **objective fitness measure** is defined here as a static function that accepts an approximation to the solution concept, and returns a value that is maximal if and only if the approximation is a member of the solution concept. A binary indicator function that returns 1 for solutions and 0 otherwise satisfies this definition, but is clearly unhelpful as a means to guide an evolutionary algorithm. To be useful in guiding an evolutionary algorithm, an objective fitness measure should furthermore provide a *gradient* towards the solution concept. It may do so by providing an estimate of the distance to the solution concept.

4.1.1 An objective fitness measure for the MEU solution concept

The MEU solution concept specifies that individuals whose average outcome against all possible tests is maximal are solutions. Thus, a straightforward choice of an objective fitness measure for the MEU solution concept is an individual's average outcome against all possible tests:

$$f_o^{MEU}(C) = \frac{\sum\limits_{T \in \mathbb{T}} G(C,T)}{|\mathbb{T}|} \tag{1}$$

where $\mathbb{T}$ is the set of all tests and $G(C,T)$ denotes the outcome of the interaction between candidate solution C and test T. This measure clearly satisfies the formal requirement that individuals that maximize the objective fitness measure should be solutions according to the solution concept and vice versa. Furthermore, by measuring the fraction of tests solved by an individual a clear gradient is obtained expressing the degree to which an individual approximates the solution concept.

4.2 Subjective fitness measures

A **subjective fitness measure** is an evaluation function used by a coevolutionary algorithm. Coevolutionary algorithms may employ a wide variety of evaluation methods; any of these qualify as subjective fitness measures. A typical example of a coevolutionary evaluation measure is the average outcome of an individual against a current population of opponents or tests. However, many other criteria can be used; for example, for any algorithm that ranks individuals in order to perform selection, the ranks of the individuals can be used as a subjective fitness measure. Whatever criterion is used for selection by a given coevolutionary algorithm is the subjective fitness measure of that algorithm.

4.3 Objective Fitness Correlation

The Objective Fitness Correlation (OFC) of a population in a coevolutionary algorithm is defined as the correlation between the objective fitness and the subjective fitness of the individuals in the population:

DEFINITION 1. *[Objective Fitness Correlation] Let $f_o(x)$ be an* objective fitness measure *for a solution concept S, and let $f_s(x)$ be the* subjective fitness measure *employed by an algorithm A, as defined above. Then the* **Objective Fitness Correlation** *$OFC(P)$ of algorithm A with respect to $f_o(x)$ for a given list P of individuals, such as a population, is defined as the correlation coefficient between the outcomes of the objective and subjective fitness measures for the individuals in P:*

$$OFC(P) = r(f_o(P), f_s(P))$$

where $f_o(P)$ and $f_s(P)$ are vectors whose i^{th} element contain the result of applying the indicated function to the i^{th} element of P, and r denotes the correlation coefficient, also known as Pearson's correlation or the product-moment coefficient, which is obtained by dividing the covariance between the two measures by the product of their standard deviations.

5. TEST PROBLEM: LINT

To test the notion of Objective Fitness Correlation, we compare the behavior of three different algorithms on an abstract test problem for which objective fitness measures can be calculated. First we discuss the requirements that motivate the choice of the test problem.

We are interested in the ability of coevolutionary algorithms to produce accurate estimates of the objective fitness of individuals. To study this ability, giving accurate estimates of this objective quality should not be trivial. Thus, depending on the current population, it must be possible for the subjective coevolutionary fitness to give misleading information about the global objective quality of individuals. A further requirement is that influences of further complicating factors, other than the property that the subjective fitness can be misleading, should be ruled out as much as possible.

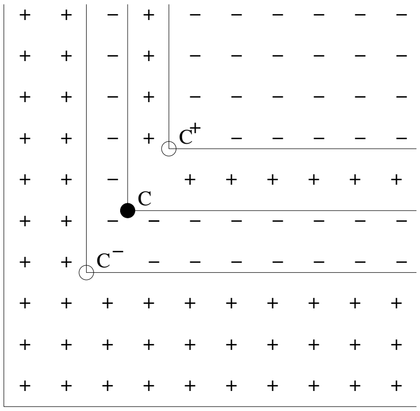

Figure 1: Diagrammatic illustration of the LINT (Locally INTransitive) problem. Plus/minus symbols ('+/-') indicate tests passed/not passed by candidate C. Given individual C, a value of ΔC_i is added in every dimension to obtain point C^+, or subtracted to obtain C^-. The basic principle of the game is that the candidate passes all tests except those that are greater or equal in all dimensions, i.e. on upper-righthand side of C. Between the points C^- and C^+ however, the outcome of this basic rule is reversed. Due to this reversal, the game is misleading on a local scale; while on a global scale the objective fitness increases by moving higher up along any of the dimensions, on a local scale the candidate may have to decrease its values to increase its subjective fitness by winning against local tests.

A test problem that satisfies both requirements is the LINT (Locally INTransitive) problem. LINT is a Numbers Game [2] that was introduced by Richard Watson [16], and has been previously used in coevolution [17]. Individuals in LINT are points in an n-dimensional space. Higher values in each dimension correspond to better performance. However, for opponents within a certain neighborhood of the individual, outcomes are reversed: lower values yield higher outcomes. Thus, while on a global scale higher values are preferable, on a local scale lower values can lead to higher performance. This misleading aspect may drive individuals towards the origin of the space.

We use a variant of the LINT problem where the size of the neighborhood is dependent on the location of the individual: for higher values, the size of the neighborhood within which the outcomes of opponents are reversed grows. This variant of LINT can be defined as follows; see Figure 1 for an illustration. Given a candidate solution C with coordinates C_i, the neighborhood of C for which outcomes are reversed is defined by two points C^+ and C^- that are obtained by adding or subtracting a factor ΔC_i in each dimension i, with $\Delta < 1$:

$$
\begin{aligned}
C_i^+ &= C_i + \Delta C_i \\
C_i^- &= C_i - \Delta C_i
\end{aligned}
$$

Given these points, the LINT problem can be defined as follows:

Definition 2 (LINT).

$$
LINT(C,T) = \begin{cases} 1 & \text{if } \forall i \; T_i \geq C_i \;\wedge\; \neg\,(\forall i \; Ti \geq C_i^+) \\ & \text{or } \quad \neg\,(\forall i \; T_i \geq C_i^-) \\ 0 & \text{otherwise} \end{cases} \qquad (2)
$$

A number of phenomena exist that can cause failure in coevolutionary algorithms. These phenomena include overspecialization (the tendency to focus on a subset of the underlying objectives) and disengagement (the tendency for individuals to become too far separated in competence, resulting in a loss of gradient); see e.g. [2, 1, 3, 18] for a discussion of these phenomena. To avoid possibly complicating effects other than inaccurate evaluation, on which we want to focus here, a one-dimensional version of the LINT game is used. Furthermore, to avoid disengagement, a single population setup is used.

5.1 Analytical determination of objective fitness for the LINT problem

For the LINT test problem, the objective fitness measure can be determined analytically, thus obviating the need to play against all tests in the domain. The objective fitness function for the MEU solution concept, see Eq. 1, specified the objective fitness of an individual C as the average outcome over all possible tests. Since the outcome against a test is either 1 (win) or 0 (lose), this figure is equal to the fraction of tests against which C wins. This in turn is equal to the fraction or volume of the search space containing tests against which C wins. A general analytical expression for the objective fitness of a LINT candidate solution C can be obtained by calculating the differences in volume of the nested hyper-rectangles spanned by on the lower-lefthand side respectively the origin and the points C^-, C and C^+, and on the upper-righthand side the upper-right hand point of the space:

$$
f_{o,LINT}^{MEU}(C) = \frac{1}{\Pi_i \; max_i} \quad (\Pi_i \; max_i - \Pi_i \; (max_i - C_i^-) + \Pi_i \; (max_i - C_i) - \Pi_i \; (max_i - C_i^+))
$$

where max_i is the maximum coordinate in dimension i and Π denotes the product operator. For the 1-dimensional version of LINT, this amounts to $\frac{C_1}{max_1}$, as can be seen directly from the diagram. For the 2-dimensional version, assuming identical horizontal and vertical limits $max_i = max$ of the space, this works out to

$$
\frac{m(C_1 + C_2) - (1 + 2\Delta^2)C_1C_2}{max^2}
$$

```
 1. $pop := initialize\_random();$
 2. for $gen = 1 : generations$ {
 3.   $pop := pop \cup mutate(pop);$
 4.   for $i = 1 : |pop|$ {
 5.       $solution\_score_i := \sum_{j=1}^{|pop|}(G(pop_i, pop_j);$
 6.       $test\_score_i := informativeness(pop, pop_i) -$
          $\sum_{C \in pop} G(C, pop_i);$
 7.     $SF_i := \alpha\ solution\_score_i + (1-\alpha)\ test\_score_i$
 8.   }
 9.   $pop := select(pop, SF);$
10. }
```

Figure 2: Pseudo-code for the Informative algorithm.

```
 1. $archive := initialize\_random();$
 2. $pop := initialize\_random();$
 3. for $genno = 1 : generations$ {
 4.   $pop := pop \cup mutate(pop, archive);$
 5.   $updateArchive();$
 7.   for $i = 1 : |pop|$ {
 8.       $solution\_score_i :=$
        $\sum_{j=1}^{|archive|}(G(pop_i, archive_j);$
 9.       $test\_score_i := informativeness(archive \cup$
        $pop, pop_i) - \sum_{C \in archive \cup pop} G(C, pop_i)$
10.     $SF_i := \alpha\ solution\_score + (1-\alpha)\ test\_score$
11.   }
12.   $pop := select(pop, SF);$
13. }
```

Figure 3: Pseudo-code for the Archive algorithm.

6. ALGORITHMS

The algorithms used in this comparison are as follows. **Basic** is a basic coevolution method using the average outcome against all population members. **Informative** takes the informativeness of tests into account. **Archive** couples coevolution to a simple archive. The algorithms are described in detail in the pseudocode of Figures 2 and 3, which is explained below.

The Basic method is identical to the Informative algorithm except for line 7, which for the Basic method reads:

$$test - score_i = -\sum_{C \in pop} G(C, pop_i);$$

The functions used in the algorithm are now described.

initialize_random() generates n individuals. In each dimension, values are uniformly distributed between 0 and 0.2.

mutate(pop) generates a new set of individuals based on the current population using mutation. n individuals are drawn randomly with replacement and mutated in all dimensions by adding a normally distributed random value with mean $mutation\_bias$ =-0.005 and standard deviation $mutation\_distance$ =0.02. Thus, mutation has a bias towards decreasing the current value in each dimension rather than increasing it. This choice is made to model the property of realistic problems that generating a mutant is more likely to decrease the objective fitness of an individual than to increase it. For the Archive method, the mutate function includes the $\frac{n}{2}$ individuals most recently added to the archive to the set of individuals to which mutation is applied.

evaluate(): All individuals in the population are played either against the population (basic and informative method) or, for the archive method, against the archive. The subjective fitness consists of two components: the solution-score and the test-score, reflecting two related roles of individuals in a coevolutionary algorithm: improving performance (reflected by the outcomes of individuals against tests), and providing an informative opponents for the evaluation of other individuals. Since a single population setup is used, individuals are evaluated on both roles.

For the non-archive methods, the solution score is calculated as the average outcome against all individuals the population. For the archive method, the solution score is the average outcome against all individuals in the archive alone; one of the goals of an archive is to provide a more stable basis for evaluation than the population.

The test-score is calculated by playing the population against either the population or, for the archive method, the population and the archive. Since this score evaluates individuals on their value as a test, the individual is used as the second player (T, in equation 2) in the interaction. Since G gives the outcome of the first player, outcomes are negated when individuals are evaluated as tests.

To measure the informativeness of an individual when viewed as a test, the *distinctions* made by each test are determined. A test T makes a distinction between two candidate solutions $C1$ and $C2$ if it assigns a higher outcome to one than to the other, for more information see [19, 1]:

$$dist(T, C1, C2) \iff G(C1, T) > G(C2, T)$$

Measuring the number of distinctions made by a test discards much information about *which* distinctions are made. To promote diversity, Competitive Fitness Sharing [20] is used: each distinction is assigned a weight of 1 over the number of tests that make the distinction. The informativeness of a test is the weighted sum of the distinctions it makes. For the informative and archive methods, the sum of the outcomes obtained as a test or second player is added to the informativeness to yield the test-score. For the basic method, informativeness is not used, and the sum of outcomes is used by itself as the test-score. For all methods, the subjective fitness of an individual is calculated as α solution-score + $(1-\alpha)$ test-score.

Select() sorts individuals based on their subjective fitness SF_i, and randomly selects n individuals with replacement using the sorting rank as the relative probability of selection.

updateArchive() randomly select a candidate solution C from the population. The outcomes of C against the archive and against the current population are determined. If no existing individual in the archive has the outcome vector, the individual is added to the archive. If the archive exceeds its maximum size MAX-ARCHIVE-SIZE, only the most recent MAX-ARCHIVE-SIZE individuals are retained.

7. EXPERIMENTS

The three algorithms described in the previous section are applied to the LINT test problem. The parameters of the experiments are as follows. A 1-dimensional version of the LINT game with parameters $\max_i = 10$ and $\Delta = 0.05$ is used. The population size $n = 20$. The relative weight of the solution-score versus the test score $\alpha = 0.9$. The number of generations GENERATIONS=500. For the archive method, MAX-ARCHIVE-SIZE=50. For each of the three experiments 50 runs are performed.

8. RESULTS

8.1 OFC of unbiased populations

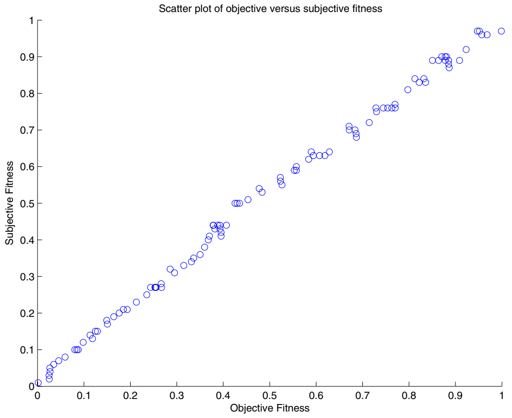

Figure 4: Scatter plot of objective versus subjective fitness.

A first observation is that for a random sample of individuals, such as the initial populations of a coevolutionary algorithm, the subjective fitness measure SF1 should provide an unbiased estimate of the objective fitness OF; this is because the average outcome against a uniform random sample of the search space provides an unbiased estimate of the average outcome against the whole search space.

To test this prediction, we generated two sets of $n = 100$ individuals, each selected uniformly randomly from the complete search space. All individuals in the first set were played against all individuals in the second set, resulting in an $n \times n$ outcome matrix. For each individual in the first set, the subjective fitness SF1 was calculated as the average of its 100 outcomes against individuals in the second set, and furthermore the objective fitness OF of all individuals in the first set was calculated. A scatter plot of the resulting relation between objective and subjective fitness is shown in Figure 4. The correlation is very high, as predicted; the correlation coefficient has a value of .9979, and thereby empirically confirms the prediction.

8.2 Objective fitness

Figure 5 shows the objective fitness over evolutionary time, averaged over 50 runs, with error bars showing the standard deviation. From the figure, it is seen that the Basic method peters out relatively early, after which the objective no longer increases, and even begins to slope down towards the end of the runs.

The Informative method performs slightly better; it continues to increase over a longer time-span, and achieves a higher final performance. However, at the end of the runs, progress has also almost completely come to a halt.

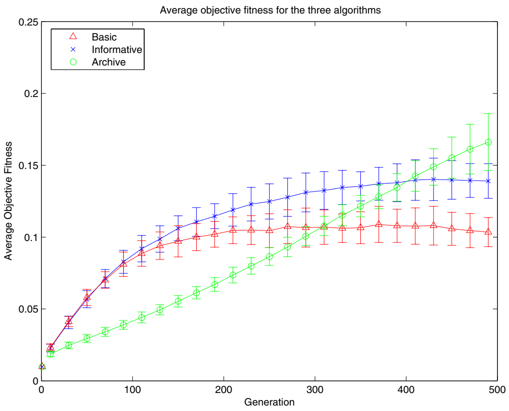

Figure 5: Objective fitness of the three methods over time.

While the addition of informativeness does improve the performance compared to the Basic algorithm, both methods are unable to achieve sustained progress. We explain this by the nature of the test problem; since the size of the neighborhood within which outcomes are reversed grows as individuals move higher on the axis, it becomes increasingly difficult to maintain a population that extends beyond this growing neighborhood. Thus, the locally intransitive nature of the problem plays an increasingly disruptive role, thereby taking away the potential for further improvement.

The curve for the Archive method displays a strikingly different behavior. While the rate of progress is lower than for the other two methods, progress is more stable and persistent. This results in a significantly higher final performance.

To test the statistical significance of the results, a Wilcoxon rank sum test was performed comparing the distribution of the 50 end-of-run values for the objective fitness of each of the three methods. It is found that the Archive method outperforms the Informative method, and the Informative method outperforms the Basic method. Both results are highly significant, with $p < 0.001$.

8.3 OFC

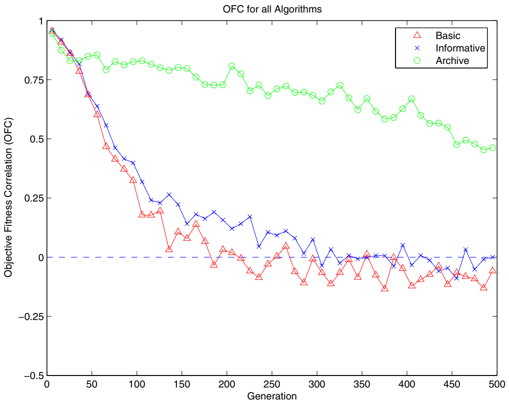

Figure 6: OFC of the methods as a function of time.

Since our aim is to study the accuracy of evaluation and its relation to performance, the OFC of the different methods has been measured over time. Figure 6 shows the result.

All methods start out with a high OFC. This is to be expected, as randomly initialized initial populations have an expected OFC of 1, as discussed in Section 8.1. Due to the biasing effect of coevolutionary selection, these high initial OFC values decrease at the beginning of a run.

Observing the OFC directly provides an explanation of the different behavior of the three methods. First, the Informative method has a slightly higher OFC, and thus a slightly more accurate evaluation, than the Basic method; this difference is reflected in the correspondingly slight performance difference.

A more striking observation is that the archive method maintains much higher OFC values than the remaining two methods, even though this value also decreases over time; the latter observation is unavoidable, as the archive is of limited size, and thus only provides a limited ability to overcome the misleading local information characterizing LINT.

A further observation to be explained is why the Archive method improves slower than the other two methods, especially given that it achieves more accurate evaluation. Since the solution score is based on the archive, the method strongly depends on inclusion of newly generated test into the archive to detect that newly generated candidate solution achieve a higher performance. While evaluating individuals against the population results in increasingly inaccurate information, differences in the performance of individuals can be detected more frequently as the population typically contains more opponents near newly generated individuals that are to be compared than the archive.

The above results demonstrate that the OFC can provide valuable information about coevolution methods that can be used to develop our understanding of the behavior of this intriguing family of algorithms. Furthermore, it has been seen that for the problem at hand, differences in the OFC correspond closely to differences in performance; this indicates that the differences in evaluation accuracy impact the performance of the algorithms.

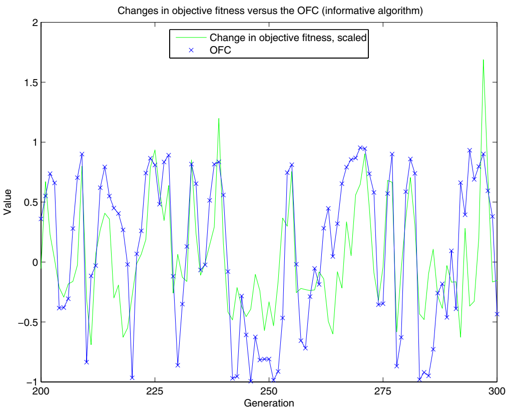

Figure 7: Relation between differences in average fitness between successive generations and the OFC: a clear relation exists, indicating the ability to achieve progress is strongly connected to the accuracy of evaluation.

To study the relation between evaluation accuracy and progress more closely, we calculate the *difference* in average objective fitness between each pair of successive generations. A positive difference indicates an increase in objective fitness from one generation to the next.

Figure 7 shows the result for the Informative method. Results are plotted for a single run, so that changes in performance can be seen in detail without the smoothing effect of aggregation. The graph provides clear evidence that changes in performance are related to the accuracy of evaluation; during periods where the OFC is high, fitness changes are almost consistently positive, and the height of the positive fitness change tends to increase. Periods with a low OFC are consistently accompanied by negative fitness changes.

9. DISCUSSION

The results in the paper may give the reader the impression that the OFC is necessarily closely tied to the objective performance of an algorithm, which might diminish its value as a new analytical tool. This impression is incorrect however; there are numerous factors that determine or influence the behavior of a coevolutionary algorithm. These include not only coevolutionary pathologies such as overspecialization and disengagement, but also different algorithmic choices such as which selection technique is used.

In the experiments, particular care has been taken to rule out the influence of factors other than the accuracy of evaluation. As a result, it has been possible to demonstrate A) that the accuracy of evaluation can indeed be measured online for actual coevolutionary algorithms, and B) that this accuracy, expressed by the OFC measure, can strongly impact the performance of the algorithm, both on a local timescale, as seen by the relation between the OFC and fitness differences, and on a global timescale, as seen by the comparison between the different algorithms. In summary, we believe the OFC is a highly useful new analytical tool for the study of coevolutionary algorithms.

The current work is restricted to abstract test problems, and calculation of the exact objective fitness is possible for a limited set of test problems only. However, as has been observed empirically, the subjective fitness of a random population corresponds almost perfectly to the objective fitness of that population. This implies that the evaluation accuracy afforded by a coevolutionary population can be estimated by playing it against a randomly generated sample of individuals; this provides a way to estimate the OFC in problems of practical interest, and may provide a useful monitoring tool.

10. CONCLUSIONS

A central question in current coevolution research is how the accuracy of coevolutionary evaluation may be improved in order to improve the efficiency and performance of this interesting class of algorithms.

Starting from the general notion of a solution concept, a new definition for the *objective fitness* in coevolution has been provided. For the particular solution concept of Maximum Expected Utility, of which maximizing the average outcomes against opponents is an example, an objective fitness function has been described. For certain problems, the objective fitness can be determined analytically, as was shown.

The main contribution is the introduction of a new analytical tool for the study of coevolution named the Objective Fitness Correlation (OFC). The OFC measure is defined as the correlation between the objective and subjective fitness of a set of individuals, such as a population. The OFC provides a precise and theoretically justified measure of the accuracy of coevolutionary evaluation.

The practical value of the OFC in analyzing coevolution algorithms has been demonstrated in experiments. Differences in evaluation accuracy exist and can actually be measured for different algorithms. Overall differences in the OFC between three algorithms corresponded to differences in objective performance. Moreover, fluctuations in the average fitness from one generation to the next were found to be strongly connected to OFC levels. This sheds new light on the dynamics of coevolutionary runs; differences that would otherwise be viewed as mere random changes are now seen to have their origin in changing levels of evaluation accuracy.

We suggest that the OFC of existing and new coevolutionary algorithms should be analyzed on test problems for which objective fitness measures are available. This approach can provide valuable information about the behavior of algorithms, and may serve to improve existing algorithms and lead to the development of new methods.

11. REFERENCES

[1] Edwin D. De Jong and Jordan B. Pollack. Ideal evaluation from coevolution. *Evolutionary Computation*, 12(2):159–192, 2004.

[2] Richard A. Watson and Jordan B. Pollack. Coevolutionary dynamics in a minimal substrate. In Spector, L., et al., editor, *Proceedings of the Genetic and Evolutionary Computation Conference, GECCO-01*, pages 702–709, San Francisco, CA, 2001. Morgan Kaufmann.

[3] Sevan G. Ficici. *Solution Concepts in Coevolutionary Algorithms*. PhD thesis, Brandeis University, 2004.

[4] Sean Luke and R. Paul Wiegand. Guaranteeing coevolutionary objective measures. In Kenneth A. De Jong, Riccardo Poli, and Jonathan E. Rowe, editors, *Foundations of Genetic Algorithms 7*, pages 237–252. Morgan Kaufmann, San Francisco, 2003.

[5] Susan L. Epstein. Toward an ideal trainer. *Machine Learning*, 15(3):251–277, 1994.

[6] Hugues Juillé and Jordan B. Pollack. Coevolving the "ideal" trainer: Application to the discovery of cellular automata rules. In John R. Koza et al., editor, *Proceedings of the Third Annual Genetic Programming Conference*, pages 519–527, San Francisco, CA, USA, 1998. Morgan Kaufmann.

[7] Anthony Bucci and Jordan B. Pollack. Order-theoretic analysis of coevolution problems: Coevolutionary statics. In *Proceedings of the GECCO-02 Workshop on Coevolution: Understanding Coevolution*, 2002.

[8] E. Popovici and Jong K. Jong. Relationships between internal and external metrics in co-evolution. In David Corne et al., editor, *Proceedings of the 2005 IEEE Congress on Evolutionary Computation*, volume 3, pages 2800–2807, Edinburgh, Scotland, UK, 2-5 September 2005. IEEE Press.

[9] D. Cliff and G. F. Miller. Tracking the Red Queen: Measurements of adaptive progress in co-evolutionary simulations. In F. Morán, A. Moreno, J. J. Merelo, and P. Chacón, editors, *Proceedings of the Third European Conference on Artificial Life: Advances in Artificial Life*, volume 929 of *LNAI*, pages 200–218, Berlin, 1995. Springer.

[10] Sevan G. Ficici and Jordan B. Pollack. Challenges in coevolutionary learning: Arms-race dynamics, open-endedness, and mediocre stable states. In C. Adami, R. Belew, H. Kitano, and T. Taylor, editors, *Proceedings of the Sixth International Conference on Artificial Life*. The MIT Press, 1998.

[11] Kenneth O. Stanley and Risto Miikkulainen. The dominance tournament method of monitoring progress in coevolution. In Alwyn M. Barry, editor, *GECCO 2002: Proceedings of the Bird of a Feather Workshops, Genetic and Evolutionary Computation Conference*, pages 242–248, New York, 8 July 2002. AAAI.

[12] Sevan G. Ficici and Jordan B. Pollack. A game-theoretic approach to the simple coevolutionary algorithm. In Schoenauer, M., et al., editor, *Parallel Problem Solving from Nature, PPSN-VI*, volume 1917 of *LNCS*, Berlin, 2000. Springer.

[13] Richard A. Watson and Jordan B. Pollack. Symbiotic combination as an alternative to sexual recombination in genetic algorithms. In Schoenauer, M., et al., editor, *Parallel Problem Solving from Nature, PPSN-VI*, volume 1917 of *LNCS*, pages 425–434, Berlin, 2000. Springer.

[14] Ari Bader-Natal and Jordan B. Pollack. Towards metrics and visualizations sensitive to coevolutionary failures. In Mitchell A. Potter and R. Paul Wiegand, editors, *Coevolutionary and Coadaptive Systems: Papers from the AAAI Fall Symposium*, Technical Report FS-05-03, pages 1–8, Menlo Park, California, 2005. AAAI Press.

[15] Edwin D. De Jong. The MaxSolve algorithm for coevolution. In Hans-Georg Beyer, editor, *Proceedings of the Genetic and Evolutionary Computation Conference, GECCO-05*, pages 483–489. ACM Press, 2005.

[16] Richard A. Watson. Personal communication, 2003.

[17] Edwin D. De Jong. Intransitivity in coevolution. In Xin Yao et al., editor, *Parallel Problem Solving from Nature - PPSN VIII*, volume 3242 of *LNCS*, pages 843–851, Birmingham, UK, 18-22 September 2004. Springer-Verlag.

[18] Anthony Bucci, Jordan B. Pollack, and Edwin D. De Jong. Automated extraction of problem structure. In *Proceedings of the Genetic and Evolutionary Computation Conference, GECCO-04*, pages 501–512, 2004.

[19] Sevan G. Ficici and Jordan B. Pollack. Pareto optimality in coevolutionary learning. In Jozef Kelemen, editor, *Sixth European Conference on Artificial Life*, pages 316–325, Berlin, 2001. Springer.

[20] Christopher D. Rosin. *Coevolutionary Search among Adversaries*. PhD thesis, University of California, San Diego, CA, 1997.