

Substitution of the Fittest: A Novel Approach for Mitigating Disengagement in Coevolutionary Genetic Algorithms

Hugo Alcaraz-Herrera^a and John Cartlidge^b

University of Bristol, Bristol, U.K.

Keywords: Coevolution, Disengagement, Genetic Algorithms.

Abstract: We propose substitution of the fittest (SF), a novel technique designed to counteract the problem of disengagement in two-population competitive coevolutionary genetic algorithms. The approach presented is domain-independent and requires no calibration. In a minimal domain, we perform a controlled evaluation of the ability to maintain engagement and the capacity to discover optimal solutions. Results demonstrate that the solution discovery performance of SF is comparable with other techniques in the literature, while SF also offers benefits including a greater ability to maintain engagement and a much simpler mechanism.

1 INTRODUCTION

While attempting the problem of designing optimal sorting networks using a genetic algorithm (GA), Hillis decided that rather than use randomly generated input lists to evaluate sorting networks, he would instead co-evolve a population of input lists that are evaluated on their ability to not be sorted (Hillis, 1990). By coupling the evolution of input lists with the evolution of the networks to sort those lists, Hillis attempted to create an “arms race” dynamic such that input lists consistently challenge networks that are sorting them. As networks improve their ability to sort, so lists become more difficult to sort, etc. This *coevolutionary* approach significantly improved results and generated wide interest amongst evolutionary computation (EC) practitioners. In particular, coevolution offers EC the ability to tackle domains where an evaluation function is unknown or difficult to operationally define; and through self-learning, coevolutionary systems offer potential for the “holy grail” of open-ended evolutionary progress.

However, it soon emerged that coevolution can suffer from some “pathologies” that cause the system to behave in an unwanted manner, and prevent continual progress towards some desired goal. For instance, coevolving populations may continually cycle with no overall progress; populations may progress in an unintended and unwanted direction; or populations may

disengage and stop progressing entirely (Watson and Pollack, 2001). These pathologies have been studied in depth and a variety of techniques have been introduced as remedy (Popovici et al., 2012). However, there is still much to be understood, and no panacea has been discovered.

Contribution: We propose *substitution of the fittest* (SF), a novel domain-independent method designed to tackle the problem of disengagement in two-population competitive coevolutionary systems. We explore and evaluate SF in the deliberately simple “greater than” domain, specifically designed for elaborating the dynamics of coevolution (Watson and Pollack, 2001). We compare performance and system dynamics against *autonomous virulence adaptation* (AVA), a technique that has been shown to reduce the likelihood of disengagement and improve optimisation in various domains (Cartlidge and Ait-Boudaoud, 2011). Initial results suggest that SF has some benefits over AVA. We evaluate and discuss the reasons why, and present avenues for future investigation.

2 BACKGROUND

Coevolutionary genetic algorithms with two distinct populations are often described using terminology that follows the biological literature. As such, and following Hillis’ original formulation, the populations are often named as “hosts” and “parasites” (Hillis, 1990). In such cases, the host population tends to

^a  <https://orcid.org/0000-0002-9991-662X>

^b  <https://orcid.org/0000-0002-3143-6355>

denote the population of candidate “solutions” that we are interested in optimising (e.g., the sorting networks), while the parasite population tends to denote the population of test “problems” for the solution population to solve (e.g., the lists to sort); i.e., the hosts are the *models* and the parasites are the *training set*; or alternatively the hosts are the *learners* and the parasites are the *teachers*. Throughout this paper, we tend to use the host-parasite terminology to distinguish coevolving populations. However, while this terminology is meaningful in *asymmetric* systems where one population (the model) is of most interest, it should be noted that in *symmetric* systems, such as games of self-play where both coevolving populations are models with the same encoding scheme, the two populations become interchangeable and the names *host* and *parasite* have less meaning.

In an ideal scenario, two-population competitive coevolution will result in an *arms race* such that both populations continually evolve beneficial adaptations capable of outperforming competitors. As a result, there is continual system progress towards some desired optimum goal. However, this ideal scenario rarely materialises. In practice, coevolutionary systems tend to exhibit pathologies that restrict progress (Watson and Pollack, 2001). These include *cycling*, where populations evolve through repeated trajectories like players in an endless game of *rock-paper-scissors*; and while short-term evolution exhibits continual progress, there is no long-term global progress (Cartlidge and Bullock, 2004b). Alternatively, populations may start to *overspecialise* on sub-dimensions of the game, such that evolved solutions are brittle and do not generalise (Cartlidge and Bullock, 2003). Furthermore, one population may begin to dominate the other to such an extent that populations *disengage* and evolutionary progress fails altogether, with populations left to drift aimlessly (Cartlidge and Bullock, 2004a). The likelihood of suffering from these pathologies can be exacerbated by the problem domain. Cycling is more likely when the problem exhibits *intransitivity*; overspecialisation is more likely in *multi-objective* problems; and disengagement is more likely if the problem has an asymmetric *bias* that favours one population (Watson and Pollack, 2001).

Numerous techniques have been proposed for mitigating the pathologies that prevent continual coevolutionary progress (for detailed reviews, see (Popovici et al., 2012; Miguel Antonio and Coello Coello, 2018)). We can roughly group these approaches into three broad categories; although in practice many techniques straddle more than one category.

First, there are *archive* methods, which are designed to preserve potentially valuable adaptations

from being “lost” during the evolutionary process. The first coevolutionary archiving technique is the *Hall of Fame* (HoF) (Rosin and Belew, 1997). Every generation, the elite member of each population is stored in the HoF archive. Then, individuals in the current population are evaluated against current competitors and also against members of the HoF. This ensures that later generations are evaluated on their capacity to beat earlier generations as well as their contemporaries. However, as the archive grows each generation, simple archiving methods like the HoF can become unwieldy over time. To counter this, more sophisticated and efficient archiving methods have been introduced to simultaneously minimise archive size while maximising archive “usefulness”. An efficient example is the Layered Pareto Coevolutionary Archive (LAPCA), which only stores individuals that are *non-dominated* and *unique*; while the archive itself is pruned over time to keep the size within manageable bounds (de Jong, 2007). More recent variations on Pareto archiving approaches include rIPCA, which has been applied to the problem of network security through the coevolution of adversarial network attack and defence dynamics (Garcia et al., 2017). Pareto dominance has also been employed for selection without the use of an archive, for example the Population-based Pareto Hill Climber (Bari et al., 2018); and Pareto fronts have been incorporated into an “extended elitism” framework, where offspring are selected only if they Pareto dominate parents when evaluated against the *same* opponents (Akinola and Wineberg, 2020).

A second popular class of approaches attempt to maintain a diverse set of evolutionary challenges through the use of *spatial embedding* and *multiple populations*. Spatially embedded algorithms – where populations exist on an n-dimensional plane and individuals only interact with other individuals in the local neighbourhood – have been shown to succeed where other non-spatial coevolutionary approaches fail. Explanations for how spatial models can help combat disengagement through challenge diversity have been explored in several works (Wiegand and Sarma, 2004; Williams and Mitchell, 2005). Challenge diversity can also be maintained through the use of multiple genetically-distinct populations (i.e., with no interbreeding or migration). Examples include the *friendly competitor*, where two *model* populations (one “friendly” and one “hostile”) are coevolved against one *test* population (Ficici and Pollack, 1998). Tests are rewarded if they are both easy to be defeated by a friendly model and hard to be beaten by a hostile model; thereby ensuring pressure on tests to evolve at a challenge-level consistent with the abil-

ity of models. Recently, a new method incorporating the periodic spawning of sub-populations, and then re-integration of individuals that perform well across multiple sub-populations back into the main population has been shown to encourage continual progress in predator-prey robot coevolution (Simione and Nolfi, 2021).

Finally, there are approaches that focus on adapting the *mechanism for selection* such that individuals are not selected in direct proportion to the number of competitions that they win; i.e., selection favours individuals that are *not* unbeatable. An early endeavour in this area is the *phantom parasite*, which marginally reduces the fitness of an unbeatable competitor, while all other fitness values remain unchanged (Rosin, 1997). Later, the Φ function was introduced for the density classification task to coevolve cellular automata rules that classify the density of an initial condition (Pagie and Mitchel, 2002). The Φ function translates all fitness values such that individuals are rewarded most highly for being equally difficult and easy to classify (i.e., by being classified correctly half of the time); while individuals that are always classified or always unclassified are punished with low fitness. However, while Φ worked well, it was limited by being domain-specific.

More generally applicable is the *reduced virulence* technique (Cartlidge and Bullock, 2002; Cartlidge and Bullock, 2004a). Inspired by the behaviour of biological host-parasite systems, where the virulence of pathogens evolves over time, reduced virulence is the first domain-independent technique with tunable parameters that can be configured. After generating a parasite score through competition, reduced virulence applies the following non-linear function to generate a fitness for selection:

$$f(x_i, \nu) = \frac{2x_i}{\nu} - \frac{x_i^2}{\nu^2} \quad (1)$$

where $0 \leq x_i \leq 1$ is the relative (or subjective) aptitude of individual i and $0.5 \leq \nu \leq 1$ represents the virulence of the parasite population. When $\nu = 1$, equation (1) preserves the original ranking of parasites (i.e., the ranking of competitive score, x) and is equivalent to the canonical method of rewarding parasites for all victories over hosts. When $\nu = 0.5$, equation (1) rewards maximum fitness to parasites that win exactly half of all competitions. Therefore, in domains where there is a bias in favour of one population (the “parasites”), setting a value of $\nu < 1$ for the advantaged population reduces the bias differential in order to preserve coevolutionary engagement. Reduced virulence demonstrated improved performance, but is limited by requiring ν to be determined in advance. In many domains, bias may be

difficult to determine and may change over time. To tackle this problem, reduced virulence has been incorporated into a human-in-the-loop system enabling a human controller to *steer* coevolution during runtime by observing the system behaviour and altering the value of ν in real time (Bullock et al., 2002).

Later, *autonomous virulence adaptation* (AVA) – a machine learning approach that automatically updates ν during coevolution – was proposed (Cartlidge and Ait-Boudaoud, 2011). Each generation t , AVA updates ν using:

$$\nu_{t+1} = \nu_t + \Delta_t \quad (2)$$

$$\Delta_t = \mu\Delta_{t-1} + \alpha(1 - \mu)(\tau - \bar{X}_t) \quad (3)$$

where $0 \leq \alpha, \mu, \tau \leq 1$ are learning rate, momentum, and target value, respectively; and \bar{X}_t is the normalised mean subjective score of the population.¹ Rigorous calibration of AVA settings demonstrated that values $\alpha = 0.0125$, $\mu = 0.3$, and $\tau = 0.56$ can be applied successfully in a number of diverse domains. In particular, it was shown that AVA can coevolve high performing sorting networks and maze navigation agents with much greater computational efficiency than archive techniques such as LAPCA (Cartlidge and Ait-Boudaoud, 2011).

3 SUBSTITUTION OF THE FITTEST

We introduce substitution of the fittest (SF), a novel technique designed to combat disengagement that is domain-independent and requires no calibration. Disengagement tends to occur when one population “breaks clear” of the competing population such that all individuals in the leading population outperform all individuals in the trailing population. Therefore, in simple terms, SF is designed to apply a “brake” to the population evolving more quickly; while for the population trying to keep pace, SF applies an “acceleration”. Consequently, the advantage of the leading population over the trailing population is reduced. In this way, SF is designed to keep populations engaged.

Unlike standard evolutionary algorithms, where individuals are evaluated using an “absolute” fitness function that is exogenous to the evolutionary process, competitive coevolutionary GAs utilise a “relative” (or “subjective”) fitness evaluation, where fitness Ψ_i of an individual i is endogenously assigned

¹For the initial $t < 5$ generations, to avoid immediate disengagement in cases of extreme bias differential, equation (3) is replaced by $\Delta_t = (0.5 - \bar{X}_t)/t$; so virulence can immediately adapt to high ($\nu = 1$) or low ($\nu = 0.5$) values.

based on performance against other evolving individuals. Usually, score ψ_i is simply the proportion of “victories” that i secures across a series of competitions against evolving opponents. These competitive interactions between coevolving populations describe a coupled system that has potential to develop into an arms-race of continual progress. However, when the populations decouple, i.e., when disengagement occurs, all information regarding the relative differences in performance of individuals is lost, such that $\forall i, j : \psi_i = \psi_j$. This is problematic and causes the coevolving populations to drift.

The current state of a population can be measured by the *population mean subjective aptitude*:

$$\sigma_{pop} = \frac{\sum_{i=1}^n \psi_i}{n} \quad (4)$$

where n is the number of individuals in the population and $0 \leq \sigma \leq 1$. Then, σ values for each population can be used to measure the *level of disengagement*, defined as:

$$\delta = |\sigma_{popA} - \sigma_{popB}| \quad (5)$$

where $0 \leq \delta \leq 1$. When populations have similar σ , then disengagement δ has a low value close to zero. When populations are fully disengaged, i.e., when $\sigma_{popA} = 1$ and $\sigma_{popB} = 0$, or when $\sigma_{popA} = 0$ and $\sigma_{popB} = 1$, then $\delta = 1$. During the coevolutionary process, δ_t is calculated and stored each generation t . If $\delta_{t+1} \leq \delta_t$, disengagement *did not* increase; otherwise, disengagement *did* increase and so SF is triggered.

When SF is triggered, we first calculate the number of individuals to be substituted, κ , defined as:

$$\kappa = n\delta^{\frac{1}{\delta}} \quad (6)$$

where n is the number of individuals in the population and the result is rounded up to the nearest integer. The value of κ increases non-linearly as a function of δ . As populations approach disengagement (δ near 1), the number of substitutions tends to n . It is important to point out that if $\kappa > \frac{n}{2}$, then effectively only $n - \kappa$ substitutions occur. For instance, given a population with $n = 6$ parasites whose aptitudes are [0.8,0.6,0.4,0.2,0.1,0.0] and $\kappa = 4$, after the substitution, their aptitudes will be [0.0,0.1,0.2,0.4,0.1,0.0]. In this example, half the population was not modified and the two fittest individuals were substituted by the two worst individuals. Furthermore, if $\delta = 1$, populations would not be modified as the number of individuals to be substituted is the same as the population size (i.e., $\kappa = n$). On the contrary, when $\delta < 0.3$, κ tends to 0. During these times, populations are sufficiently engaged and substitutions are not necessary.

The next step in the SF process consists of comparing σ_{popA} and σ_{popB} and then substituting κ individuals in each population using the following rules:

- *Population with lowest σ* : Rank all individuals by subjective aptitude ψ_i . Then, the κ individuals with the *highest* ψ_i replace the κ individuals with the *lowest* ψ_i . Finally, the subjective aptitude of every individual is increased by:

$$\psi'_i = \min(\psi_i + \delta, 1) \quad (7)$$

taking minimum value to ensure $0 \leq \psi'_i \leq 1$.

- *Population with highest σ* : Rank all individuals by ψ_i . Then κ individuals with the *lowest* ψ_i replace the κ individuals with the *highest* ψ_i ; following replacement, the subjective aptitude of every individual is decreased using:

$$\psi'_i = \max(\psi_i - \delta, 0) \quad (8)$$

taking maximum value to ensure $0 \leq \psi'_i \leq 1$.

As described, SF affects each population in a different manner. For the population that evolves more quickly, the proportion of individuals to be randomly selected is increased. For instance, if the highest ranked individual whose $\psi < \delta$, then the κ individuals would be selected at random because they would have $\psi' = 0$. On the other hand, for the population which evolves slower than the other, the effect is the opposite, i.e., if the lowest ranked individual whose $\psi + \delta \geq 1$, then all κ individuals would have $\psi' = 1$ and hence those individuals would have high probability of being selected.

Following SF, selection is performed and genetic operators are applied as usual. In the minimal experiments we present in the following sections, we use tournament selection and apply mutation, i.e., populations are asexual and recombination is not used.

4 EXPERIMENTAL METHOD

4.1 The “Greater Than” Game

The *greater than* game (Watson and Pollack, 2001) was introduced as a minimal (and analytically tractable) substrate capable of demonstrating some of the pathological dynamics of coevolution; in particular *disengagement*. The game consists of maximizing scalar values through a comparison-based function where given two scalar values, α and γ , the function operates as $score(\alpha, \gamma) = 1$ if $\alpha > \gamma$, 0 otherwise. Here, we use a slightly modified “greater than or equals” game that rewards draws; such that $score(\alpha, \gamma) = 1$ if $\alpha > \gamma$; 0.5 if $\alpha = \gamma$; 0 otherwise.

The coevolutionary set-up consists of two isolated populations, each with n individuals. Each individual is represented by a binary string with $l = 100$ bits and

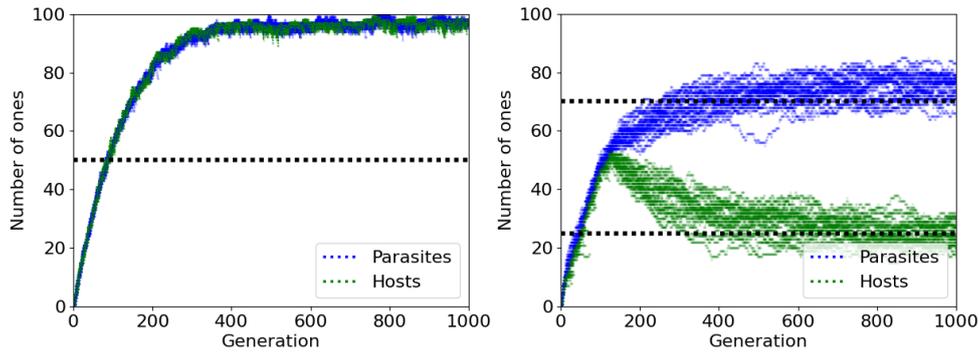


Figure 1: Coevolution: (left) equal bias $\beta_h = \beta_p = 0.5$; and (right) differential bias $\beta_h = 0.25$, $\beta_p = 0.75$.

the “objective” purpose of the coevolutionary system is to evolve individuals with bit-strings containing all ones (i.e., scalar values of $l = 100$). Every generation, to generate a subjective aptitude score, each individual plays the greater than game against a sample of S opponents. Tournament selection is used to select individuals for reproduction, and the only genetic operator is mutation (i.e., reproduction is *asexual*). Mutation has a bias controlled by parameter β , where $0 \leq \beta \leq 1$. For each bit, there is a probability m of mutation occurring. When it occurs, the bit is assigned a new value at random, with probability β of assigning a 1, and probability $1 - \beta$ of assigning a 0; i.e., when $\beta = 0.5$, mutation has an equal chance of assigning the bit to 1 or 0; when $\beta = 0$ mutation will always assign the bit to 0; and when $\beta = 1$ mutation will always assign the bit to 1. This bias parameter β allows the simple game to emulate the intrinsic asymmetry of real and more complex domains, where it is often easier for one population to outperform another. Under mutation bias alone, i.e., when populations are disengaged and left to drift under the absence of selection pressure, we expect the population to tend towards having $\beta \times l$ ones. Therefore, for a bias $\beta = 0.5$, we would expect the population to drift towards scalar values of 50.

In our two-population competitive set up, we label the populations as *hosts* and *parasites*. Each population has an independent bias value β , which controls the problem difficulty for each population. We use β_h to label the bias value of the host population and β_p to label the bias value of the parasite population. When bias differential is high, i.e., when the value of β_p is much larger than the value of β_h (or vice versa), disengagement becomes more likely as the game is much easier for the parasites (alternatively, the hosts) to succeed. In more complex domains, there is often asymmetry in problem difficulty for coevolving populations. By varying β_p and β_h , we are able to control the asymmetry of problems in the simple greater than game.

Our experimental set up is detailed as follows. We coevolve two isolated populations, each with 25 individuals ($n = 25$). The length of the binary array (an individual) is 100 ($l = 100$) and each bit is initialised to 0. For generating a competitive score, we use a sample size of 5 ($S = 5$). Tournament size for selection is 2. The probability of mutation per bit is 0.005 ($m = 0.005$). Finally, each evolutionary run lasts for 1000 generations.

4.2 Disengagement

The effect of disengagement can most clearly be elaborated by visualising what happens when it occurs. Figure 1 (left) presents the coevolution of two populations, each with equal bias $\beta_p = \beta_h = 0.5$. We see that populations remain engaged throughout the evolutionary run. This engagement provides a continual gradient for selection and encourages an “arms race” of increasing performance. As a result, both populations reach optimal performance of 100 ones. This is far higher than 50 ones (dotted line) that both populations would be expected to reach if drifting through space under mutation alone, i.e., when selection pressure is removed.

In contrast, Figure 1 (right) demonstrates the pathology of disengagement. Here, there is a differential bias in favour of parasites, such that $\beta_p = 0.75$ and $\beta_h = 0.25$. Initially, both populations remain engaged and selection drives evolutionary progress, with both populations reaching approximately 50 ones by generation 150. However, the impact of differential bias in favour of parasites then leads to a disengagement event such that all parasites have more ones than their competing hosts; resulting in a subjective score of zero for all hosts and a subjective score of one for all parasites. At this point selection pressure is removed as all individuals have an equal (and therefore random) chance of selection, leaving the populations to drift under mutation alone. As expected, the parasite population drifts to the parasite mutation

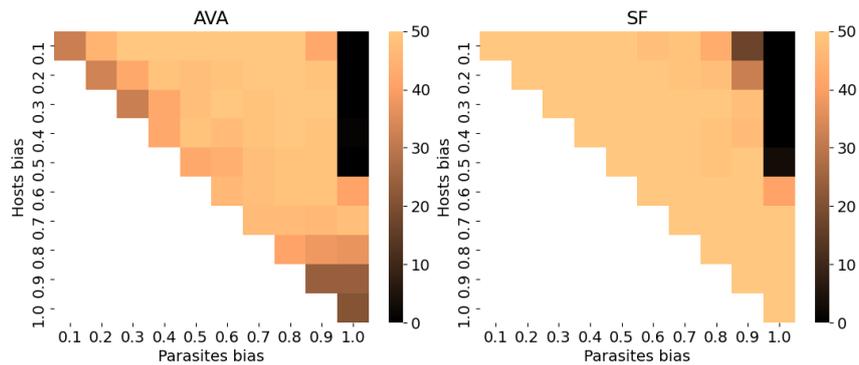


Figure 2: Number of runs with no disengagement; AVA (left) and SF (right) across all bias levels (50 trials).

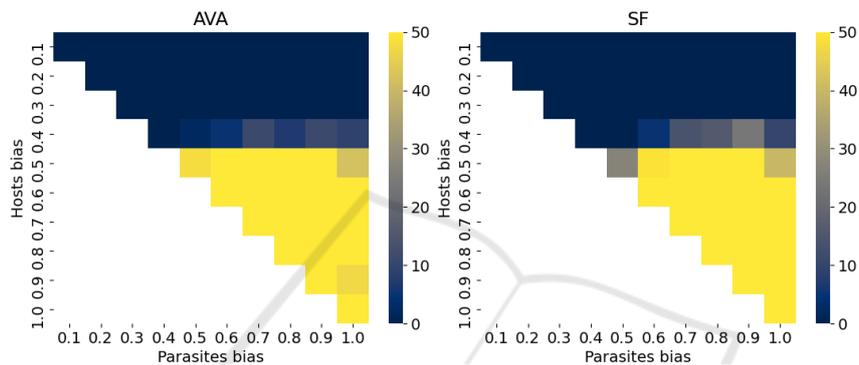


Figure 3: Number of runs where hosts reached optimum; AVA (left) and SF (right) across all bias levels (50 trials).

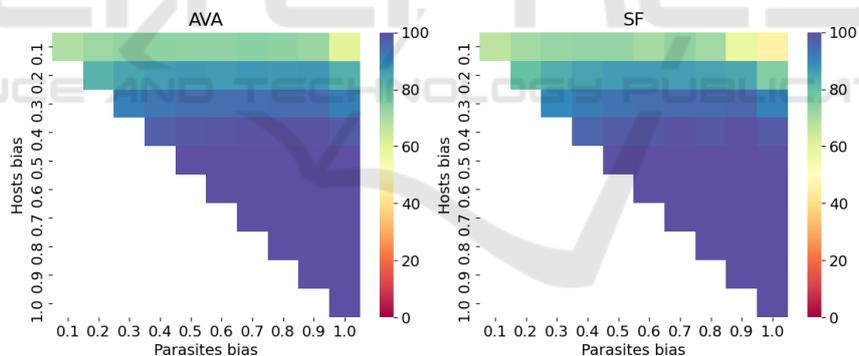


Figure 4: Mean number of ones of best host; AVA (left) and SF (right) across all bias levels (50 trials).

bias (dotted line) of 75, while hosts degrade to the host mutation bias of 25. The high bias differential between populations ($\beta_p - \beta_h = 0.5$) not only causes the initial disengagement event, but ensures that post-disengagement populations drift through different regions of genotype/phenotype space and will never re-engage through chance alone.

In general, the greater the bias differential between populations, the greater the likelihood of disengagement occurring. As shown, disengagement severely hinders coevolutionary progress.

5 SF VS AVA: A COMPARISON

To measure the performance of SF, we perform a thorough comparison against AVA (Cartlidge and Ait-Boudaoud, 2011), which has been previously shown to dramatically reduce the effects of disengagement in the greater than game and also in several more complex and realistic domains, including designing minimal-length sorting networks and discovering classifier systems for maze navigation. To understand how the two approaches are likely to perform in more complex domains, where population asymmetries are

more likely, we trial both SF and AVA in simulations where mutation bias is varied across all possible levels ($\beta_p \geq \beta_h$) in the range [0.1, 1.0]. The mutation bias was configured in favour of parasites (except when $\beta_p = \beta_h$); although this decision is arbitrary and results where bias is in favour of hosts would yield symmetrically similar results. For each bias scenario, we performed 50 experimental trials. To analyse performance of SF and AVA, we utilise three metrics: (i) the reliability of the technique to maintain population engagement; (ii) the capacity to discover optimal hosts containing all ones; and (iii) the mean number of ones that hosts reach before disengagement occurs. The following sections describe our findings.

5.1 Maintaining Engagement

Since the main objective of AVA and SF is to maintain engagement during coevolution, regardless of bias fluctuations that may arise, it is fundamental to study their response under diverse bias levels. Figure 2 presents a heatmap showing the number of runs where AVA and SF maintained population engagement during the full coevolutionary process (regardless of whether an optimal host is found).

Overall, we see that SF is able to maintain engagement more successfully than AVA across a diverse range of bias differentials. For 38 of the 50 bias pairings, SF maintains engagement for the full coevolutionary run of 1000 generations across all 50 trials. For AVA, however, this number is only 12. Moreover, SF maintains engagement for the full coevolutionary run in at least 40 out of 50 trials across 49 bias pairings, while for AVA this number is 42.

Relative to SF, results suggest that AVA tends to struggle in scenarios (i) where the bias of both populations are either the same (symmetrical systems) or similar; and (ii) where parasite bias is very high (e.g., $\beta_p = 1.0$) and there is a large bias differential between parasites and hosts. In comparison, although SF also fails in scenarios where parasites have very high bias, SF is capable of maintaining engagement where AVA is not.

Regarding the result obtained by AVA, in the original research (Cartlidge and Ait-Boudaoud, 2011), it was calibrated to only handle bias levels in the range [0.5, 1.0]. Moreover, in the original experiments the number of generations is 750 whereas in these experiments, the number of generations is 1000. The duration of experimental trials is a key factor inasmuch AVA, in a number of bias scenarios, tends to allow disengagement *after* optimal hosts are found. For instance, when $\beta_h = 0.5, \beta_p = 1.0$, populations tend to first reach the optimum, but then later, around genera-

tions 850-900, the populations disengage. This unexpected behaviour suggests that AVA parameters may require recalibration to maintain engagement over long time periods when there is a high bias differential. It also demonstrates an advantage of SF over AVA, as SF has no parameter settings to calibrate.

5.2 Reaching the Optimum

Another essential aspect to analyse is the capability to reach the optimal zone. Figure 3 presents the number of runs where hosts (more precisely, at least one host) reached the optimal, regardless of whether or not populations disengage after this point. We see a similar pattern for both SF and AVA. AVA reached the optimal zone at least 40 times under 21 bias levels, where as SF was capable of reaching 40 times or more under 20 bias levels. Furthermore, both AVA and SF reached the optimal zone a maximum of 50 times (i.e., every time) under 18 bias scenarios.

As expected, in multiple bias scenarios hosts were not capable of reaching the optimum when hosts have a very low mutation bias ($\beta_h < 0.5$).

5.3 General Performance

Figure 4 shows the mean maximum number of ones of the best host across all bias configurations, regardless of whether or not populations disengage or hosts reach the optimum. Again, performance of SF and AVA is similar. AVA reaches at least 90 ones under 36 bias scenarios, whereas SF reaches 90 ones in 35 scenarios. Furthermore, AVA and SF both reach 100 ones under 18 possible bias scenarios.

Results suggest that AVA and SF tend to behave similarly across most bias levels. However, when there is a significant bias differential (e.g., $\beta_h = 0.1, \beta_p = 0.9$), AVA enables populations to reach a greater performance (closer to the optimum) than SF.

5.4 Coevolutionary Dynamics

Figures 5 and 6 present example runs to highlight the effects of AVA (left) and SF (right) during the coevolutionary process. When bias differential is relatively large (Figure 5; $\beta_h = 0.3$ and $\beta_b = 0.7$) both AVA and SF maintain engagement throughout. However, when bias differential is very large (Figure 6; $\beta_h = 0.2$ and $\beta_b = 1.0$), disengagement occurs under both techniques, but tends to occur earlier (around generation 410 for AVA, compared with generation 550 for SF).

Interestingly, in Figure 6 we see that SF induces different population dynamics with respect to AVA. Under SF, in the early generations, parasites exhibit

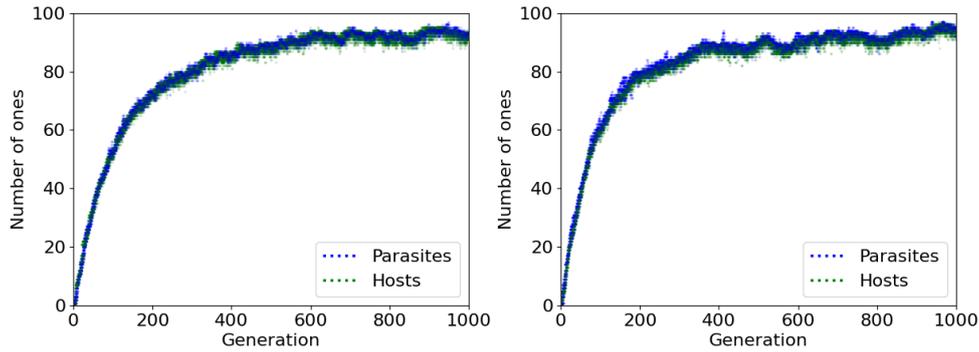


Figure 5: Example coevolutionary runs under AVA (left) and SF (right) with $\beta_h = 0.3$, $\beta_p = 0.7$.

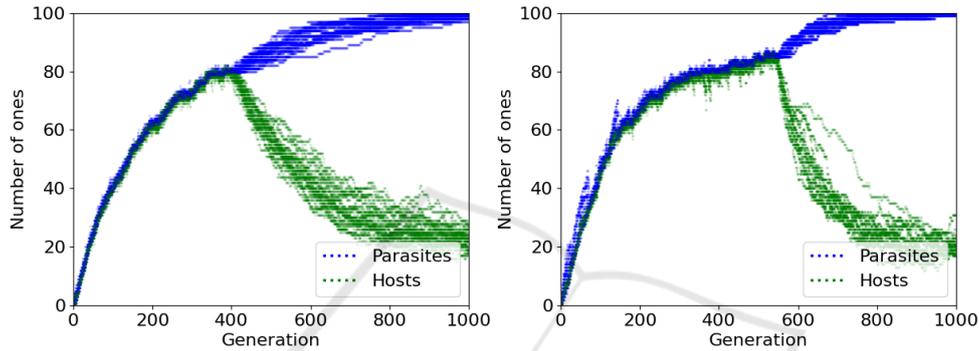


Figure 6: Example coevolutionary runs under AVA (left) and SF (right) with $\beta_h = 0.2$, $\beta_p = 1.0$.

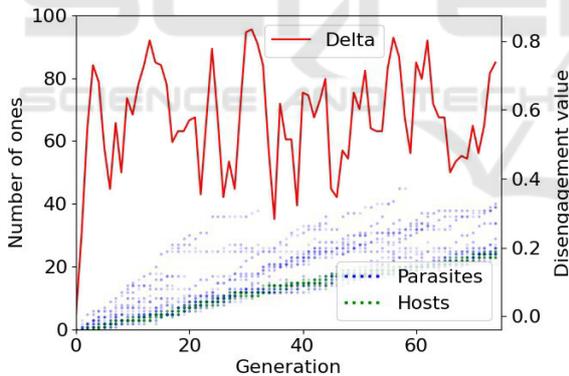


Figure 7: Initial generations of one run ($\beta_h = 0.1$, $\beta_p = 1.0$) using SF, showing absolute fitness of parasites and hosts (left axis) and δ value of disengagement (right axis).

lots of variation, with some outliers drifting far from the engaged populations. Then, around generations 90 and 170, these outlier lineages suddenly disappear. A similar “cull” effect is *not* observed with AVA.

To further investigate the cull effect in SF and its direct relation with disengagement value δ , Figure 7 presents the initial generations of one example run with very high bias differential ($\beta_h = 0.1$, $\beta_p = 1.0$). Around generation 30, we see that there is a single outlier parasite containing around 40 ones. As a consequence, $\delta \approx 0.81$ and $\kappa = 25(0.81^{1/0.81}) = 19$ (see

Equation 6). Since $19 > \frac{25}{2}$, the effective substitutions are 6 (see example given in Section 3). Therefore, the 6 fittest parasites, including the outlier, are substituted by the 6 worst parasites. Consequently, the subsequent generations do not present outliers and δ (and therefore κ) decreases. Other outlier lineages later begin to emerge and the process repeats. In contrast, AVA tends to keep both populations more tightly coupled throughout.

6 CONCLUSIONS

This research has introduced SF as an alternative technique to mitigate disengagement in competitive coevolutionary genetic algorithms. Using a minimal problem domain to enable exposition, we compared the performance of SF with AVA, a technique in the literature that has been shown to combat disengagement in a variety of domains. Experimental results suggest that, in general, SF has similar performance to AVA in terms of discovery of optimal solutions. However, SF is also shown to have better performance than AVA in terms of consistently maintaining engagement across a wide variety of bias differentials, i.e., where there is a large inherent advantage in favour of one co-evolving population. The mechanism of SF is deliber-

ately designed to be simple and domain independent, requiring no domain knowledge or specific calibration. This makes SF more easy to implement than other techniques and offers the possibility of being more generally applicable.

However, one of the potential weaknesses observed in SF is the highly-fluctuating behaviour induced in populations (i.e., the “cull” effect), which might lead to sudden disengagement in other more realistic domains. Thus, we believe that SF deserves further exploration; although it has shown suitable performance in a simple domain, experiments in more complex domains such as maze navigation or sorting networks (Cartlidge and Ait-Boudaoud, 2011) are necessary to demonstrate its reliability. Furthermore, a robust comparison against other state-of-art techniques will be performed. Finally, we intend to explore the effects that SF has on other coevolutionary pathologies, such as overspecialisation and cycling.

ACKNOWLEDGEMENTS

Hugo Alcaraz-Herrera’s PhD is supported by The Mexican Council of Science and Technology (Consejo Nacional de Ciencia y Tecnología - CONACyT). John Cartlidge is sponsored by Refinitiv.

REFERENCES

- Akinola, A. and Wineberg, M. (2020). Using implicit multi-objectives properties to mitigate against forgetfulness in coevolutionary algorithms. In *Genetic and Evolutionary Computation Conference*, GECCO, pages 769–777.
- Bari, A. G., Gaspar, A., Wiegand, R. P., and Bucci, A. (2018). Selection methods to relax strict acceptance condition in test-based coevolution. In *Congress on Evolutionary Computation*, CEC, pages 1–8.
- Bullock, S., Cartlidge, J., and Thompson, M. (2002). Prospects for computational steering of evolutionary computation. In *Workshop Proceedings 8th Int. Conf. on Artificial Life*, ALIFE, pages 131–137.
- Cartlidge, J. and Ait-Boudaoud, D. (2011). Autonomous virulence adaptation improves coevolutionary optimization. *IEEE Transactions on Evolutionary Computation*, 15(2):215–229.
- Cartlidge, J. and Bullock, S. (2002). Learning lessons from the common cold: How reducing parasite virulence improves coevolutionary optimization. In *Congress on Evolutionary Computation*, CEC, pages 1420–1425 vol.2.
- Cartlidge, J. and Bullock, S. (2003). Caring versus sharing: How to maintain engagement and diversity in coevolving populations. In Banzhaf, W., Ziegler, J., Christaller, T., Dittrich, P., and Kim, J. T., editors, *Advances in Artificial Life*, pages 299–308, Berlin, Heidelberg. Springer Berlin Heidelberg.
- Cartlidge, J. and Bullock, S. (2004a). Combating coevolutionary disengagement by reducing parasite virulence. *Evolutionary Computation*, 12(2):193–222.
- Cartlidge, J. and Bullock, S. (2004b). Unpicking tartan CIAO plots: Understanding irregular coevolutionary cycling. *Adaptive Behavior*, 12(2):69–92.
- de Jong, E. D. (2007). A monotonic archive for pareto-coevolution. *Evolutionary Computation*, 15(1):61–93.
- Ficici, S. G. and Pollack, J. B. (1998). Challenges in coevolutionary learning: Arms-race dynamics, open-endedness, and mediocre stable states. In *International Conference on Artificial Life*, ALIFE, pages 238–247.
- Garcia, D., Lugo, A. E., Hemberg, E., and O’Reilly, U.-M. (2017). Investigating coevolutionary archive based genetic algorithms on cyber defense networks. In *Genetic and Evolutionary Computation Conference Companion*, GECCO, pages 1455–1462.
- Hillis, W. D. (1990). Co-evolving parasites improve simulated evolution as an optimization procedure. *Phys. D*, 42(1–3):228–234.
- Miguel Antonio, L. and Coello Coello, C. A. (2018). Coevolutionary multiobjective evolutionary algorithms: Survey of the state-of-the-art. *IEEE Transactions on Evolutionary Computation*, 22(6):851–865.
- Pagie, L. and Mitchel, M. (2002). A comparison of evolutionary and coevolutionary search. *International Journal of Computational Intelligence and Application*, 2(1):53–59.
- Popovici, E., Bucci, A., Wiegand, R. P., and De Jong, E. D. (2012). Coevolutionary principles. In *Handbook of Natural Computing*, pages 987–1033. Springer Berlin Heidelberg, Berlin, Heidelberg.
- Rosin, C. D. (1997). *Coevolutionary Search Among Adversaries*. PhD thesis, Department of Computer Science, University of California, San Diego, California.
- Rosin, C. D. and Belew, R. K. (1997). New methods for competitive coevolution. *Evolutionary Computation*, 5(1):1–29.
- Simione, L. and Nolfi, S. (2021). Long-term progress and behavior complexification in competitive coevolution. *Artificial Life*, 26(4):409–430.
- Watson, R. A. and Pollack, J. B. (2001). Coevolutionary dynamics in a minimal substrate. In *Genetic and Evolutionary Computation Conference*, GECCO, pages 702–709.
- Wiegand, R. P. and Sarma, J. (2004). Spatial embedding and loss of gradient in cooperative coevolutionary algorithms. In *Parallel Problem Solving from Nature*, PPSN, pages 912–921.
- Williams, N. and Mitchell, M. (2005). Investigating the success of spatial coevolution. In *Genetic and Evolutionary Computation Conference*, GECCO, pages 523–530.