

NASA/TM-2001-210930



Improving Simulated Annealing by Replacing its Variables with Game-Theoretic Utility Maximizers

David Wolpert
Ames Research Center, Moffett Field, California

Esfandiar Bandari
RIACS, Ames Research Center, Moffett Field, California

Kagan Tumer
Ames Research Center, Moffett Field, California

National Aeronautics and
Space Administration

Ames Research Center
Moffett Field, California 94035-1000

September 2001

Available from:

NASA Center for Aerospace Information
7121 Standard Drive
Hanover, MD 21076-1320
(301) 621-0390

National Technical Information Service
5285 Port Royal Road
Springfield, VA 22161
(703) 487-4650

Improving Simulated Annealing by Replacing its Variables with Game-Theoretic Utility Maximizers

David H. Wolpert
NASA Ames Research Center
MS 269-1
Moffett Field, CA 94035
dhw@ptolemy.arc.nasa.gov

Esfandiar Bandari
NASA Ames Research Center
MS 269-1
Moffett Field, CA 94035
bandari@mail.arc.nasa.gov

Kagan Tumer
NASA Ames Research Center
MS 269-3
Moffett Field, CA 94035
kagan@ptolemy.arc.nasa.gov

Abstract

The game-theory field of Collective INtelligence (COIN) concerns the design of computer-based players engaged in a non-cooperative game so that as those players pursue their self-interests, a pre-specified global goal for the collective computational system is achieved “as a side-effect”. Previous implementations of COIN algorithms have outperformed conventional techniques by up to several orders of magnitude, on domains ranging from telecommunications control to optimization in congestion problems. Recent mathematical developments have revealed that these previously developed algorithms were based on only two of the three factors determining performance. Consideration of only the third factor would instead lead to conventional optimization techniques like simulated annealing that have little to do with non-cooperative games. In this paper we present an algorithm based on all three terms at once. This algorithm can be viewed as a way to modify simulated annealing by recasting it as a non-cooperative game, with each variable replaced by a player. This recasting allows us to leverage the intelligent behavior of the individual players to substantially improve the exploration step of the simulated annealing. Experiments are presented demonstrating that this recasting significantly improves simulated annealing for a model of an economic process run over an underlying small-worlds topology. Furthermore, these experiments reveal novel small-worlds phenomena, and highlight the shortcomings of conventional mechanism design in bounded rationality domains.

1 INTRODUCTION

There are three general types of distributed systems that are found in nature and that researchers have translated into computational algorithms for function maximization. The first is exemplified

by Neo-Darwinian natural selection, which has been translated into genetic algorithms (GA's) [1, 7, 13, 24]. These distributed systems can be viewed as finding a maximum of a function G that takes as argument any single one of the system's variables. (Each of those single variables is a "genome", with G of a genome being the "fitness" of the "phenotype" induced by that genome.)

Whereas systems of this first type have a "narrow G ", in the second type of distributed system, the function G being optimized is "wide", taking the state of the entire distributed system as its argument. In some such distributed systems it is only in the crudest sense that the individual variables can be viewed as players in a non-cooperative game. These systems comprise the second type, and examples include simulated annealing (SA [18, 11]) and swarm intelligence [2, 20], inspired by spin relaxation in physics and eusocial insect colonies, respectively.

In the third type of system, G is also wide, but the value of each of the individual variables going into G is set by a player engaged in an over-arching non-cooperative game where each player η is trying to maximize its associated payoff utility function g_η . Roughly speaking, such collective systems work when the utility functions of the individual variables/players are all "aligned" with the world utility G . Under these circumstances, as the individual players pursue their self-interests, the global goal for the full collective of maximizing G is achieved "as a side-effect". The primary naturally-occurring instances of such collectives are economic institutions where the players are human beings, e.g., auctions and clearing of markets. In the computational versions of such systems the players are instead computer programs [4, 5, 6, 12, 15, 17, 19, 30, 36, 48].

The "Collective INtelligence" (COIN) framework concerns the design of such collectives involving non-cooperative games. In particular, it addresses the issue of how to generate, from a provided G , the set of utilities $\{g_\eta\}$ that have optimal signal/noise for each player η while also having the property that as the individual players maximize those utilities, G gets maximized (i.e., while also being "aligned with G "). This work on design of collectives can be viewed as an extension of mechanism design [10] beyond human economics, to include concern for the signal-to-noise ratio in the payoff functions and off-equilibrium behavior, to allow far more freedom in choice of the g_η than exists with human players (for example to encompass computational systems in which the issue of incentive compatibility is moot), and also to encompass arbitrary G and arbitrary dynamics of the system. Applications of this framework on problems from routing in telecommunication networks [37, 44, 46] to congestion problems [47] have resulted in substantial performance improvement over conventional techniques that do not consider issues like signal-to-noise. Typically as the size of the collective grows, such improvements reach several orders of magnitude.

Recent mathematical developments have shown that the previously developed COIN algorithms for design of collectives were based on only two of the three factors determining performance at maximizing G . Consideration of only the third factor would instead lead to conventional wide- G systems that have little to do with non-cooperative games, like simulated annealing. Consideration of all three terms at once therefore would result in an algorithm that combines the two types of wide- G function maximization systems, those having naturally-occurring analogues of human economics and of statistical physics, respectively.

In this paper we present such a hybrid algorithm. Because of the similarity of this algorithm to (certain aspects of) how human corporations are run, we call it the Computational Corporation (CoCo) algorithm. Roughly speaking, it works by modifying the exploration step

of simulated annealing by having the new values of the variables be set by the moves of intelligent players in a non-cooperative game rather than by sampling a probability distribution. Like simulated annealing, the computational corporation algorithm is intended not to give the best possible performance in all problem domains — an algorithm laboriously tailored for a particular domain will invariably perform best for that domain [45]. Rather like other algorithms related to naturally-occurring distributed function maximizers the computational corporation algorithm is intended as a powerful and broadly applicable “off-the-shelf” algorithm.

In other work we present experiments demonstrating that the computational corporation algorithm outperforms simulated annealing by several orders of magnitudes for spin glass relaxation and bin-packing [43]. Here we present such experiments for a simple economics model of a set of people choosing among various potential formats for their home music-reproduction systems. In this model the players interacted over an underlying ring-like network. The world utility was the sum of each player η 's “happiness” function, which depended on factors like which of η 's neighbors picked the same format it did, η 's intrinsic preference for the format it picked, and the price, set by the level of global demand, of any reproduction in that format. In our experiments we compared CoCo to SA and also to a variant of a conventional COIN algorithm that is similar to the economics/mechanism design technique of providing incentives to players that “endogenize their externalities”. It was found that when that network had only short-range connections, the performance of CoCo at a certain iteration substantially exceeded that of the economics-like COIN, verifying the sub-optimality of such conventional economics incentive schemes (world utility equals 68 and 60, respectively, in arbitrary units, at a fixed time-step). In turn, both that COIN and CoCo system (the two game-theory-based systems) performed substantially better than SA ($G = 44$). Performances were also consistent with the spin-glass and bin-packing convergence results reported in [43], namely that CoCo would reach a given level of G about two orders of magnitude more quickly than SA.

We then modified the network by incorporating a few random long-range connections, to produce a small-worlds-network [21, 26, 27, 28, 39]. It was found that the resultant decrease in average inter-player distance resulted in a barely significant improvement in CoCo's performance (3%), and none in the other algorithms, contrary to typical results in the small-worlds literature. However if G was also changed, to reflect the total number of other players within a given player η 's full neighborhood (rather than just η 's nearest neighbors), then going to a small-worlds network improved performance substantially (10%). Note that the fact that all three algorithms experienced this improvement to some degree suggests that a variant of the small-worlds phenomenon generically extends beyond domains where it has previously been investigated to maximization of high-dimensional functions defined over a network.

2 The Mathematics of Collective Intelligence

The full formalization of the COIN framework extends significantly beyond what is needed for this paper.¹ The restricted version needed here starts with an arbitrary vector space Z whose

¹That framework encompasses, for example, arbitrary dynamic redefinitions of the players (i.e., dynamic reassignments of how the various subsets of the variables comprising the collective are assigned to players), as well as modification of the players' information sets (i.e., modification of inter-player communication). See [42].

elements ζ give the joint state of all the variables in the collective.

We wish to search for the ζ that maximizes the provided **world utility** G . In addition to G we are concerned with **payoff utility functions** $\{g_\eta\}$, one such function for each variable/player η . We use the notation $\hat{\eta}$ to refer to all players other than η .

We will need to have a way to “standardize” utility functions so that the numeric value they assign to a ζ only reflects their ranking of ζ relative to certain other elements of Z . We call such a standardization of utility U for player η the “**intelligence** for η at ζ with respect to U ”. Here we will use intelligences that are equivalent to percentiles:

$$\epsilon_U(\zeta : \eta) \equiv \int d\mu_{\zeta_\eta}(\zeta') \Theta[U(\zeta) - U(\zeta')], \quad (1)$$

where the subscript on the (normalized) measure indicates it is restricted to ζ' sharing the same non- η components as ζ , and where the Heaviside function Θ is defined to equal 1 when its argument is greater than or equal to 0, and to equal 0 otherwise. Intelligence values are always between 0 and 1.

Our uncertainty concerning the behavior of the system is reflected in a probability distribution over Z . Our ability to control the system consists of setting the value of some characteristic of the collective, e.g., setting the payoff functions of the players. Indicating that value by s , our analysis revolves around the following central equation for $P(G | s)$, which follows from Bayes’ theorem:

$$P(G | s) = \int d\vec{\epsilon}_G P(G | \vec{\epsilon}_G, s) \int d\vec{\epsilon}_g P(\vec{\epsilon}_G | \vec{\epsilon}_g, s) P(\vec{\epsilon}_g | s), \quad (2)$$

where $\vec{\epsilon}_g \equiv (\epsilon_{g_{\eta_1}}(\zeta : \eta_1), \epsilon_{g_{\eta_2}}(\zeta : \eta_2), \dots)$ is the vector of the intelligences of the players with respect to their associated payoff functions, and $\vec{\epsilon}_G \equiv (\epsilon_G(\zeta : \eta_1), \epsilon_G(\zeta : \eta_2), \dots)$ is the vector of the intelligences of the players with respect to G .

Note that $\epsilon_{g_\eta}(\zeta : \eta) = 1$ means that player η is fully rational at ζ , in that its move maximizes its payoff, given the moves of the players. So a Nash equilibrium is a point ζ where $\epsilon_{g_\eta}(\zeta : \eta) = 1$ for all players η .² On the other hand, a ζ at which all components of $\vec{\epsilon}_G = 1$ is a local maximum of G (or more precisely, a critical point of the $G(\zeta)$ surface).

If we can choose s so that the third conditional probability in the integrand is peaked around vectors $\vec{\epsilon}_g$ all of whose components are close to 1, then we have likely induced large (payoff function) intelligences. If we can also have the second term be peaked about $\vec{\epsilon}_G$ equal to $\vec{\epsilon}_g$, then $\vec{\epsilon}_G$ will also be large. Finally, if the first term in the integrand is peaked about high G when $\vec{\epsilon}_G$ is large, then our choice of s will likely result in high G , as desired.

Intuitively, the requirement that payoff functions have high “signal-to-noise” (an issue not considered in conventional work in mechanism design) arises in the third term. It is in the second term that the requirement that the payoff functions be “aligned with G ” arises. Previously

²Consideration of points ζ at which not all intelligences equal 1 provides the basis for a model-independent formalization of bounded rationality game theory, a formalization that contains variants of many of the theorems of conventional full-rationality game theory. See [41].

developed COIN algorithms concentrated on these two terms. In contrast, non-game-theory function maximization techniques like simulated annealing instead address how to have term 1 have the desired form. They do this by trying to ensure that the local maxima that the underlying system settles on have high G .

It is the simultaneous concern for all three of the terms in Eq. 2 that underlies the CoCo algorithm. To present that algorithm we must first review some COIN results on how to simultaneously set terms 2 and 3 to have the desired form.

Our desired form for the second term in Eq. 2 is assured if the collective be **factored**, which means that $\bar{\epsilon}_g$ equals $\bar{\epsilon}_G$ exactly for all ζ . In game-theory language, the Nash equilibria of a factored collective are local maxima of G . In addition to this desirable equilibrium behavior, factored collectives also automatically provide appropriate off-equilibrium incentives to the players.

As a trivial example, any “team game” in which all the payoff functions equal G is factored [9, 25]. However team games often have very poor forms for term 3 in Eq. 2, forms which get progressively worse as the size of the collective grows. This is because for such payoff functions each player η will usually confront a very poor “signal-to-noise” ratio in trying to discern how its actions affect its payoff $g_\eta = G$, since so many other player’s actions also affect G and therefore dilute η ’s effect on its own payoff function.

Previous COIN algorithms were based on varying the payoff functions $\{g_\eta\}$ to optimize the signal/noise ratio reflected in the third term, subject to the requirement that the system be factored. To understand how those algorithms work, given a measure $d\mu(\zeta_\eta)$, define the **opacity** at ζ of utility U as

$$\Omega_U(\zeta : \eta, s) \equiv \int d\zeta' J(\zeta' | \zeta) \frac{|U(\zeta) - U(\zeta'_\eta, \zeta_\eta)|}{|U(\zeta) - U(\zeta_\eta, \zeta'_\eta)|}, \quad (3)$$

where J is defined in terms of the underlying probability distributions.³

The denominator absolute value in the integrand in Eq. 3 reflects how sensitive $U(\zeta)$ is to changing ζ_η . In contrast, the numerator absolute value reflects how sensitive $U(\zeta)$ is to changing ζ'_η . So the smaller the opacity of a payoff function g_η , the more $g_\eta(\zeta)$ depends only on the move of player η , i.e., the better the associated signal-to-noise ratio for η . Intuitively then, lower opacity should mean it is easier for η to achieve a large value of its intelligence.

To formally establish this, we use the same measure $d\mu$ to define opacity as the one that defined intelligence. Under this choice expected opacity bounds how close to 1 expected intelligence can be [42]:

$$\begin{aligned} E(\epsilon_U(\zeta : \eta) | s) &\leq 1 - K, \text{ where} \\ K &\leq E(\Omega_U(\zeta : \eta, s) | s). \end{aligned} \quad (4)$$

³Writing it out in full, $J(\zeta' | \zeta) \equiv J(\zeta_\eta, \zeta'_\eta | \zeta_\eta, s) / P(\zeta_\eta | \zeta_\eta, s)$, with

$$J(\zeta_\eta, \zeta'_\eta | \zeta_\eta, s) \equiv \frac{P(\zeta_\eta | \zeta_\eta, s)P(\zeta'_\eta | \zeta_\eta, s)\mu(\zeta'_\eta)}{2} + \frac{P(\zeta'_\eta | \zeta'_\eta, s)P(\zeta_\eta | \zeta'_\eta, s)\mu(\zeta_\eta)}{2}.$$

So low expected opacity of utility g_η is a necessary condition for the third term in Eq. 2 to have the desired form for player η . While low opacity is not, formally speaking, a sufficient condition for $E(\epsilon_U(\zeta : \eta) | s)$ to be close to 1, in practice the bounds in Eq. 4 are usually tight.

In general it is not possible for a collective both to be factored and to have zero opacity for all of its players. However consider **difference** utilities, which are of the form

$$U(\zeta) = G(\zeta) - \Gamma(f(\zeta)) \quad (5)$$

where $\Gamma(f)$ is independent ζ_η . Any difference utility is factored [42]. In addition, under usually benign approximations, $E(\Omega_u | s)$ is minimized over the set of difference utilities by choosing

$$\Gamma(f(\zeta)) = E(G | \zeta_\eta, s), \quad (6)$$

up to an overall additive constant. We call the resultant difference utility the **Aristocrat** utility (AU), loosely reflecting the fact that it measures the difference between a player's actual action and the average action.

If possible, we would like each player η to use the associated AU as its payoff function to ensure good form for both terms 2 and 3 in Eq. 2. This is not always feasible however. The problem is that to evaluate the expectation value defining its AU each player needs to evaluate the current probabilities of each of its potential moves. However if the player then changes its payoff function to be the associated AU it will in general substantially change its ensuing behavior. (The player now wants to choose moves that maximize a different function from the one it was maximizing before.) In other words, it will change the probabilities of its moves, which means that its new payoff function is in fact not the AU for its actual (new) probabilities.

There are ways around this self-consistency problem, but in practice it is often easier to bypass the entire issue, by giving each η a payoff function that does not depend on the probabilities of η 's own moves. One such payoff function is the **Wonderful Life** Utility (WLU). The WLU for player η is parameterized by a pre-fixed **clamping element** CL_η chosen from among η 's possible moves:

$$WLU_\eta \equiv G(\zeta) - G(\zeta_\eta, CL_\eta). \quad (7)$$

WLU is factored no matter what the choice of clamping element. Furthermore, while not matching the low opacity of AU, WLU usually has far better opacity than does a team game.

In many circumstances one can loosely interpret a particular choice of clamping element for player η as equivalent to a "null" move for player η , equivalent to removing that player from the system. (Hence the name of this payoff function — cf. the Frank Capra movie.) For such a clamping element assigning the associated WLU to η as its payoff function is closely related to the economics technique of "endogenizing a player's externalities" [10]. However it is usually the case that using WLU with a clamping element that is as close as possible to the expected move defining AU results in far lower opacity than does clamping to the null move. Accordingly, use of such an alternative WLU almost always results in far better values of G than does the "endogenizing" WLU.

Typically, COINs in which the payoff functions are WLU or AU not only far outperform team games, but also conventional function maximization techniques like simulated annealing.

However note that even if the payoff functions in the COIN result in the collective’s having every component of the vector \vec{e}_G assuredly equal 1 — the best terms 2 and 3 we could hope for — nothing in Eq. 2 precludes a poor value for $G(\zeta)$. This is because having all those intelligences equal 1 only means that the collective is at a local maximum of G , not a global one.

This potential shortcoming is reflected in the first term in Eq. 2, a term that does not directly depend on the choice of the players’ payoff functions. Crudely speaking, what that term reflects is the propensity of the system to get stuck in a local maximum. Accordingly, one can use many of the conventional exploration/exploitation function maximization techniques like simulated annealing to induce a good form for that term. In this hybrid, at each iteration, the exploration step is determined by the moves chosen by the players, rather than by using one of the random sampling schemes that are traditionally employed. The exploitation step though is the same as in the traditional formulation of the algorithm. In this way all three terms of Eq. 2 will have a desired form, and the induced G should be large.

In its concern for all three terms this algorithm bears many similarities to well-run modern human corporations, with G the “bottom line” of the entire corporation, the players η identified with the employees of the corporation, and the associated g_η given by the employees’ performance-based compensation packages. For example, for a “factored corporation”, each employee’s compensation package contains incentives designed such that the better the bottom line of the corporation, the greater the employee’s compensation. In addition, if the compensation packages are “low opacity”, the employees will have a relatively easy time discerning the relationship between their behavior and their compensation. Finally, the centralized exploitation process in CoCo is similar to the centralized decision-making of upper management that tries to determine whether to abandon or stick with a particular set of behaviors by the employees. It is due to these similarities that we call this algorithm the computational corporation algorithm.

3 Details of the CoCo algorithm

In the version of simulated annealing explored here, at the beginning of the exploration step of each time-step t , every player η changed its move from the one it settled on at the end of the preceding time-step, $\zeta_{\eta,t-1}$, with probability .25. That 25% probability was uniformly allotted across all moves that differed from $\zeta_{\eta,t-1}$. Then in the exploitation step, G was evaluated for the new point and compared to $G(\zeta_{\eta,t-1})$. If the new point had a higher G , it became the final point of the current time-step, $\zeta_{\eta,t}$. Otherwise $\zeta_{\eta,t}$ was chosen to be either the new point or $\zeta_{\eta,t-1}$, according to a temperature-parameterized Boltzmann distribution based on the two associated G values.

In the CoCo variant of simulated annealing, the exploitation step was unchanged while the exploration step was modified to incorporate a COIN. Rather than have each player η pick an exploration move according to the simulated annealing distribution $h(\zeta_\eta)$, that move was picked by sampling the distribution $\frac{h(\zeta)c(\zeta)}{\sum_{\zeta'} h(\zeta')c(\zeta')}$, where the distribution $c(\zeta)$ was generated by a reinforcement learning (RL) algorithm [8, 33, 46, 47] using a COIN-based utility function g_η .

The RL algorithm used was perhaps the simplest one possible. Each player η would maintain a running average of the reward it has received for each of its possible moves. (Those averages were

formed by exponentially weighting moves according to how long ago they were taken.) Those averages were then used to specify a Boltzmann distribution (parameterized by an associated “learning temperature”) over the possible moves. That distribution was sampled to decide η ’s next move. In these experiments, to form the initial averages, for the first 100 time-steps, the moves for all players were chosen completely at random rather than via RL, and the associated rewards recorded by each of the players.

The AU version of CoCo was based on the very rough approximation that each player η had a uniform probability distribution over its possible moves. Together with our choice of an extremely crude RL algorithm (and the resultant need to dedicate time-steps to a “thrashing” training period in which G does not improve), these approximations constituted a significant handicap for AU CoCo.

4 Experimental Results

Small world networks are rings in which a few random long-range links are added. They have been studied in variety of domains, including immunology, communication networks, social sciences, neuro physiology and information exchange [21, 23, 26, 27, 28, 38, 39]. It has been found that despite the fact that there are relatively few long-range links in such networks, those links drastically reduce the average smallest number of links connecting any two nodes. Accordingly, addition of those few links results in information propagating across such networks far more quickly.

We wished to investigate whether the increased information speed accompanying those few extra links would improve the performance of a function-maximizer run on a function defined over the network, one whose maximization required coordination of the states across the entire network. In particular, we were interested in this issue when the function was an economic model, so that speeding up the function-maximization translated into more quickly settling into a desirable global economic state.

The model we decided to investigate was one of coordination of choice of music-reproduction format across the players/nodes. More precisely, each player η on the network was given a choice of four objects, and the player’s move was to choose three of those. The motivation was that η would buy music-reproduction systems for those three formats only. Accordingly, η would only buy music in those three formats. This led to a reduction in the global price for those three formats, due to economies of scale in reproduction, which meant that all players using those formats saw a lower price. In addition, η was only able to exchange music in the three formats it had chosen with its neighbors lying $\leq d$ links away. Finally, η had an intrinsic preference for each of the four formats, due to factors like their audio fidelity characteristics. These three effects combined to set each player η ’s “happiness” for a global configuration of all players’ format choices. In turn, G was the sum over all players of their happiness’:

$$G = \sum_{i=1}^{N_f} \left(\left(\sum_{j=1}^{N_a} \vartheta_{j,i} \right) \cdot \left(\sum_{j=1}^{N_a} \sum_{k \in \text{neigh}_j} \omega_{j,k,i} \text{pref}_{j,i} \right) \right) \quad (8)$$

where N_f and N_a are the numbers of formats and players, respectively; neigh_j is the set of neighbors for player j (a maximum distance d away from player j); $\text{pref}_{j,i}$ is the preference

of player j for format i , and was set randomly (to between 0 and 1) for each successive run; $\vartheta_{j,i} = 1$ if player j has accepted format i and 0 otherwise; and $\omega_{i,j,k} = 1$ if an player (j) and its neighbor (k) have accepted (and can therefore trade) the same format i , and 0 otherwise. The first parenthesis reflects the global popularity of a format in the network, while the second accounts for the popularity of a format in the local market (neighborhood) and individual player's preferences.

We compared simulated annealing, the computational corporation algorithm, and a simplified version of the standard economics approach of "endogenizing one's externalities". For CoCo, there were three variants: The first used team game utilities (CoCo -TG), the second used AU utilities with probabilities fixed to .25 for each of the four allowed moves (CoCo-AU), and the third was a Wonderful Life Utility with clamping parameter (1,1,1,1) i.e., clamping was to the nominally illegal action of choosing all formats (CoCo-WLU). The last choice was the "endogenizing"-like approach of WLU with a clamping parameter of (0,0,0,0), i.e., of choosing no formats at all (CoCo-Econ). For this last choice each player used a utility function of the marginal contribution of that player to the global utility: that player computes the difference between the world utility and the world utility when it abstains from participating in the exchange.

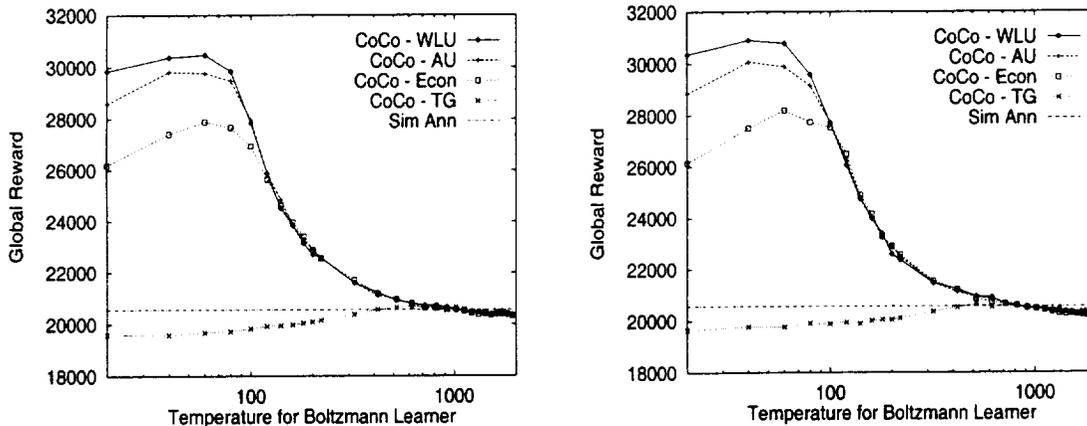


Figure 1: Small worlds network with 100 players (neighborhood size = 1): (a) Short range (left); and (b) Long range connections.

Figure 1 shows the performance at time step 200 for two variants of an underlying ring of 100 players with 6 non-nearest-neighbor links superimposed. In the first variant all the extra links were of length 2, positioned randomly. In the second extra links were purely random, giving us a small-world network. All but team game CoCo algorithms significantly outperform simulated annealing. Among those, AU CoCo and "accept-all" CoCo (WLU) also outperform the economics based approach. Also note that simulated annealing did not benefit from the existence of long range connections. However, both the different CoCo algorithms, and the economics-based algorithm showed modest improvements (3%) in the presence of the long range connections, showing that these algorithms used the new information more efficiently than did simulated annealing.

This experiment demonstrated that the existence of long range connections alone is not sufficient for the system to exhibit significant "small worlds" phenomena, and attain higher per-

formance. In particular, the model used above did not account for the “local neighborhood” (also known as path lengths [27]) which determines the distance over which a player can make a trade. Our second set of experiments incorporated this concept by modifying G to reflect the total number of other players within a given player η ’s full neighborhood (rather than just η ’s nearest neighbors). Figure 2 shows the performance of CoCo and SA in this new problem. Once again, the WLU and AU-based CoCo outperformed the economics-based CoCo, and all three outperformed SA. Also, notice that in this case, the presence of the long-range connections provided a marked improvement in the performance of all the algorithms (10%).

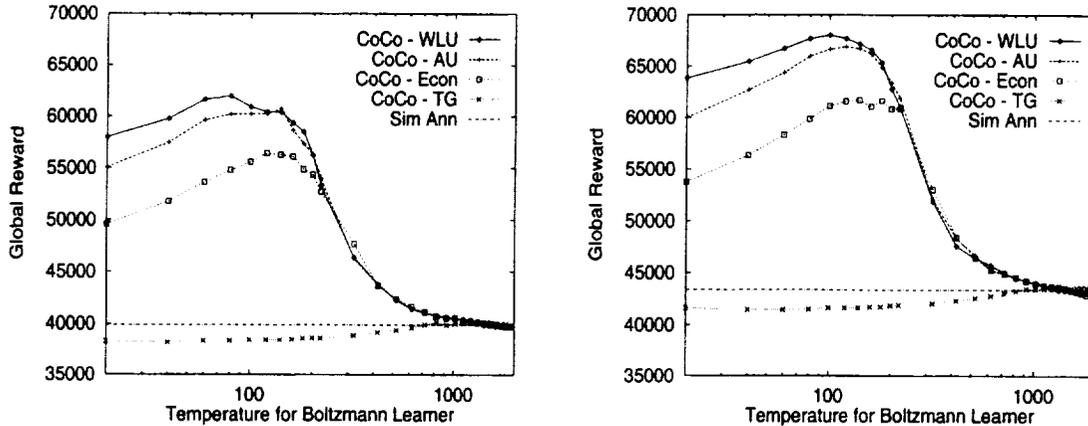


Figure 2: Small worlds network with 100 players (neighborhood size = 3): (a) Short range (left); and (b) Long range connections.

Figure 3 compares the convergence properties of CoCo WLU, CoCo-Econ and SA. The WLU-based algorithm converged to high G very rapidly, while CoCo-Econ did so more slowly. Simulated annealing, on the other hand, provided very poor G at $t = 200$. Projecting the convergence rate of SA linearly⁴ provided over two orders of magnitude slower convergence than WLU-based algorithms to good values of G . Also note the clear difference in the quality of the exploration step between CoCo and SA: the exploration step of both CoCo algorithms improves dramatically over time due to the players learning “where to explore”. SA on the other hand has minimal improvement over the exploration space, which explains its slow convergence to good G .

5 Conclusions

There are three general types of parallel systems found in nature that can be viewed as engaging in maximization of a function G . These are exemplified by neo-Darwinian natural selection (for G that take any single one of the elements of the parallel system as an argument), spin glass relaxation (for G that take the entire system as argument), and clearing of markets in economics relaxation (for G that take the entire system as argument and in which the overall parallel system can be viewed as a non-cooperative game). All three types of systems have been translated into computational algorithms, exemplified by genetic algorithms, simulated annealing,

⁴This favors SA since in practice its convergence rate slows down.

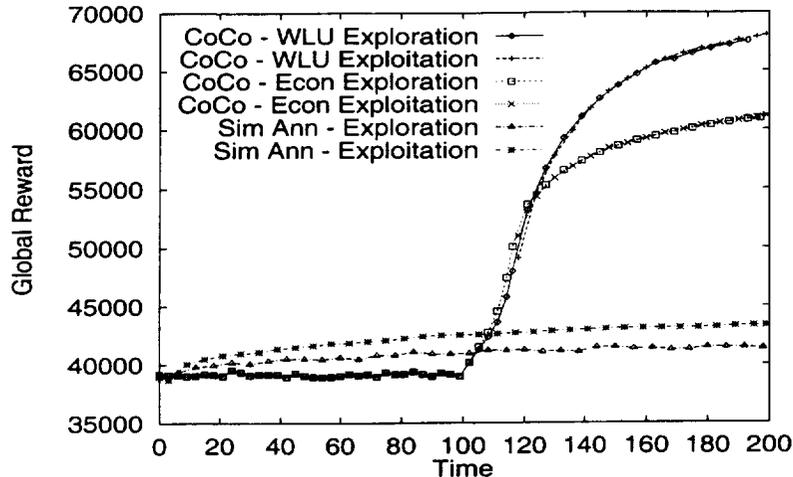


Figure 3: Small worlds network with 100 players (neighborhood size = 3).

and computational markets, respectively.

The Collective Intelligence framework can be viewed as an extension of conventional economics-based systems of the third type, to reflect signal-to-noise issues and greater freedom in modifying the individual players than exist in economies of human beings. It has traditionally been applied only to systems of the third type. Recent mathematical advances in that framework have shown that those traditional COIN algorithms only account for two of the three factors determining performance. The third factor can be accounted for by integrating the COIN with a technique of the second type, like simulated annealing. Intuitively, such an integrated system, which we call a computational corporation, can be viewed as conventional simulated annealing modified by having the value of each variable in the exploration step of the SA be set by a (computer-based) player in an associated non-cooperative game. Doing this allows the leveraging of the intelligence of such players to improve the exploration, and thereby improve the performance.

We present experiments demonstrating that the computational corporation algorithm outperforms simulated annealing by two order of magnitude for a model of an economic process run over an underlying small-worlds topology. Furthermore, these experiments reveal novel small-worlds phenomena, and highlight the shortcomings of conventional mechanism design in bounded rationality domains.

Acknowledgements: The authors thank Michael New for helpful comments.

References

- [1] T. Back, D. B. Fogel, and Z. Michalewicz, editors. *Handbook of Evolutionary Computation*. Oxford University Press, 1997.
- [2] E. Bonabeau, M. Dorigo, and G. Theraulaz. Inspiration for optimization from social insect behaviour. *Nature*, 406(6791):39–42, 2000.

- [3] C. Boutilier, Y. Shoham, and M. P. Wellman. Editorial: Economic principles of multi-agent systems. *Artificial Intelligence Journal*, 94:1-6, 1997.
- [4] G. Caldarelli, M. Marsili, and Y. C. Zhang. A prototype model of stock exchange. *Europhysics Letters*, 40:479-484, 1997.
- [5] A. Cavagna. Irrelevance of memory in the minority game. preprint cond-mat/9812215, December 1998.
- [6] D. Challet and Y. C. Zhang. On the minority game: Analytical and numerical studies. *Physica A*, 256:514, 1998.
- [7] K. Chellapilla and D.B. Fogel. Evolution, neural networks, games, and intelligence. *Proceedings of the IEEE*, pages 1471-1496, September 1999.
- [8] C. Claus and C. Boutilier. The dynamics of reinforcement learning cooperative multiagent systems. In *Proceedings of the Fifteenth National Conference on Artificial Intelligence*, pages 746-752, Madison, WI, June 1998.
- [9] R. H. Crites and A. G. Barto. Improving elevator performance using reinforcement learning. In D. S. Touretzky, M. C. Mozer, and M. E. Hasselmo, editors, *Advances in Neural Information Processing Systems - 8*, pages 1017-1023. MIT Press, 1996.
- [10] D. Fudenberg and J. Tirole. *Game Theory*. MIT Press, Cambridge, MA, 1991.
- [11] S. Geman and D. Geman. Stochastic relaxation, Gibbs distribution and the Bayesian restoration of images. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 6:721-741, 1984.
- [12] O. Guenther, T. Hogg, and B. A. Huberman. Market organizations for controlling smart matter. In *Proceedings of the International Conference on Computer Simulation and Social Sciences*, 1997.
- [13] J. H. Holland, editor. *Adaptation in Natural and Artificial Systems*. MIT Press, 1993.
- [14] J. Hu and M. P. Wellman. Online learning about other agents in a dynamic multiagent system. In *Proceedings of the Second International Conference on Autonomous Agents*, pages 239-246, May 1998.
- [15] B. A. Huberman and T. Hogg. The behavior of computational ecologies. In *The Ecology of Computation*, pages 77-115. North-Holland, 1988.
- [16] B. A. Huberman, R. M. Lukose, and T. Hogg. An economics approach to hard computational problems. *Science*, 275:51-54, 1997.
- [17] N. F. Johnson, S. Jarvis, R. Jonson, P. Cheung, Y. R. Kwong, and P. M. Hui. Volatility and agent adaptability in a self-organizing market. preprint cond-mat/9802177, February 1998.
- [18] S. Kirkpatrick, C. D. Jr Gelatt, and M. P. Vecchi. Optimization by simulated annealing. *Science*, 220:671-680, May 1983.

- [19] Y. A. Korilis, A. A. Lazar, and A. Orda. Achieving network optima using Stackelberg routing strategies. *IEEE/ACM Transactions on Networking*, 5(1):161–173, 1997.
- [20] M.J.B. Krieger, J.-B. Billeter, and L. Keller. Ant-like task allocation and recruitment in cooperative robots. *Nature*, 406:992–995, 2000.
- [21] R. V. Kulkarni, E. Almaas, and Stroud D. Exact results and scaling properties of small-world networks. *Physical Review E*, 61(4):4268–4271, 2000.
- [22] M. Lauer and M. Riedmiller. An algorithm for distributed reinforcement learning in cooperative multi-agent systems. In *Proceedings of the Seventeenth International Machine Learning Conference*, pages 535–542. Morgan Kaufman, 2000.
- [23] N. Mathias and V. Gopal. Small worlds: How and why. *Physical Review E*, 63, 2001.
- [24] M. Mitchell. *An Introduction to Genetic Algorithms*. MIT Press, Cambridge, MA, 1996.
- [25] D. Monderer and L. S. Sharpley. Potential games. *Games and Economic Behavior*, 14:124–143, 1996.
- [26] M. E. J. Newman. Models of the small world (a review). *Journal of Statistical Physics*, 101:819–841, 1987.
- [27] M. E. J. Newman, C. Moore, and D. J. Watts. Mean-field solution of the small-world network model. *Physical Review Letters*, 84(14):3201–3204, 2000.
- [28] M. E. J. Newman and D. J. Watts. Scaling and percolation in the small-world network model. *Physical Review E*, 60(6):7332–7342, 1999.
- [29] T. Sandholm and V. R. Lesser. Coalitions among computationally bounded agents. *Artificial Intelligence*, 94:99–137, 1997.
- [30] R. Savit, R. Manuca, and R. Riolo. Adaptive competition, market efficiency, phase transitions and spin-glasses. preprint cond-mat/9712006, December 1997.
- [31] A. Schaerf, Y. Shoham, and M. Tennenholtz. Adaptive load balancing: A study in multi-agent learning. *Journal of Artificial Intelligence Research*, 162:475–500, 1995.
- [32] Y. Shoham and K. Tanaka. A dynamic theory of incentives in multi-agent systems. In *Proceedings of the International Joint Conference on Artificial Intelligence*, 1997.
- [33] R. S. Sutton and A. G. Barto. *Reinforcement Learning: An Introduction*. MIT Press, Cambridge, MA, 1998.
- [34] L. Tesfatsion. How economists can get ALIFE. In Arthur *et al.*, editor, *The Economy as an Evolving System*, volume 27. Adison Wesley, 1997.
- [35] L. Tesfatsion. Agent-based computational economics: A brief guide to the literature. In J. Michie, editor, *Reader's Guide to Social Sciences*, volume 1. Fitzroy-Dearborn Wesley, London, UK, 2000.

- [36] P. Tucker and F. Berman. On market mechanisms as a software techniques. Technical Report CS96-513, University of California, San Diego, December 1996.
- [37] K. Tumer and D. H. Wolpert. Collective intelligence and Braess' paradox. In *Proceedings of the Seventeenth National Conference on Artificial Intelligence*, pages 104-109, Austin, TX, 2000.
- [38] D. Watts. *Small Worlds*. Princeton University Press, Princeton, NJ, 1999.
- [39] D. J. Watts and S. H. Strogatz. Collective dynamics of 'small world' networks. *Nature*, 393:440-442, 1998.
- [40] M. P. Wellman. A market-oriented programming environment and its application to distributed multicommodity flow problems. In *Journal of Artificial Intelligence Research*, 1993.
- [41] D. H. Wolpert. Bounded-rationality game theory. pre-print, 2001.
- [42] D. H. Wolpert. The mathematics of collective intelligence. pre-print, 2001.
- [43] D. H. Wolpert, E. Bandari, and K. Tumer. Improving simulated annealing by recasting it as a non-cooperative game. *Nature*, 2001. submitted.
- [44] D. H. Wolpert, S. Kirshner, C. J. Merz, and K. Tumer. Adaptivity in agent-based routing for data networks. In *Proceedings of the fourth International Conference of Autonomous Agents*, pages 396-403, 2000.
- [45] D. H. Wolpert and W. G. Macready. No free lunch theorems for optimization. *IEEE Transactions on Evolutionary Computation*, 1(1):67-82, 1997. Best Paper Award.
- [46] D. H. Wolpert, K. Tumer, and J. Frank. Using collective intelligence to route internet traffic. In *Advances in Neural Information Processing Systems - 11*, pages 952-958. MIT Press, 1999.
- [47] D. H. Wolpert, K. Wheeler, and K. Tumer. Collective intelligence for control of distributed dynamical systems. *Europhysics Letters*, 49(6), March 2000.
- [48] Y. C. Zhang. Modeling market mechanism with evolutionary games. *Europhysics Letters*, March/April 1998.

REPORT DOCUMENTATION PAGE

Form Approved
OMB No. 0704-0188

Public reporting burden for this collection of information is estimated to average 1 hour per response, including the time for reviewing instructions, searching existing data sources, gathering and maintaining the data needed, and completing and reviewing the collection of information. Send comments regarding this burden estimate or any other aspect of this collection of information, including suggestions for reducing this burden, to Washington Headquarters Services, Directorate for Information Operations and Reports, 1215 Jefferson Davis Highway, Suite 1204, Arlington, VA 22202-4302, and to the Office of Management and Budget, Paperwork Reduction Project (0704-0188), Washington, DC 20503.

1. AGENCY USE ONLY (Leave blank)		2. REPORT DATE August 2001	3. REPORT TYPE AND DATES COVERED Technical Memorandum	
4. TITLE AND SUBTITLE Improving Simulated Annealing by Replacing Its Parameters With Game-Theoretic Utility Maximizers			5. FUNDING NUMBERS 755-07	
6. AUTHOR(S) David Wolpert, Esfandiar Bandari, Kagan Tumer				
7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES) Ames Research Center Moffett Field, CA 94035-1000			8. PERFORMING ORGANIZATION REPORT NUMBER	
9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS(ES) National Aeronautics and Space Administration Washington, DC 20546-0001			10. SPONSORING/MONITORING AGENCY REPORT NUMBER NASA/TM-2001-210930	
11. SUPPLEMENTARY NOTES Point of Contact: Author, Ames Research Center, MS 269-1, Moffett Field, CA 94035-1000 (650) 604-3362				
12a. DISTRIBUTION/AVAILABILITY STATEMENT Unclassified — Unlimited Subject Category 59 Distribution: Standard Availability: NASA CASI (301) 621-0390			12b. DISTRIBUTION CODE	
13. ABSTRACT (Maximum 200 words) The game-theory field of Collective Intelligence (COIN) concerns the design of computer-based players engaged in a non-cooperative game so that as those players pursue their self-interests, a pre-specified global goal for the collective computational system is achieved as a side-effect. Previous implementations of COIN algorithms have outperformed conventional techniques by up to several orders of magnitude, on domains ranging from telecommunications control to optimization in congestion problems. Recent mathematical developments have revealed that these previously developed algorithms were based on only two of the three factors determining performance. Consideration of only the third factor would instead lead to conventional optimization techniques like simulated annealing that have little to do with non-cooperative games. In this paper we present an algorithm based on all three terms at once. This algorithm can be viewed as a way to modify simulated annealing by recasting it as a non-cooperative game, with each variable replaced by a player. This recasting allows us to leverage the intelligent behavior of the individual players to substantially improve the exploration step of the simulated annealing. Experiments are presented demonstrating that this recasting significantly improves simulated annealing for a model of an economic process run over an underlying small-worlds topology. Furthermore, these experiments reveal novel small-worlds phenomena, and highlight the shortcomings of conventional mechanism design in bounded rationality domains.				
14. SUBJECT TERMS collective intelligence, algorithm, optimization techniques, simulated annealing			15. NUMBER OF PAGES 18	
			16. PRICE CODE A03	
17. SECURITY CLASSIFICATION OF REPORT Unclassified	18. SECURITY CLASSIFICATION OF THIS PAGE Unclassified	19. SECURITY CLASSIFICATION OF ABSTRACT	20. LIMITATION OF ABSTRACT	

