Collective Intelligence

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

January 27, 2003

1 MOTIVATION AND BACKGROUND

1.1 Collectives

Many systems of self-interested agents have an associated performance criterion that rates the dynamic behavior of the overall system. This paper presents an introduction to the science of such systems. Formally, this paper concerns **collectives**, which are defined as any system having the following two characteristics: First, the system must contain one or more agents each of which we view as trying to maximize an associated **private utility**. Second, the system must have an associated **world utility** function that rates the possible behaviors of that overall system [38, 39, 40, 37, 28, 38]. In practice collectives are often very large, distributed, and support little if any centralized communication and control, although those characteristics are not part of their formal definition.

A naturally occurring example of a collective is a human economy. One can identify the agents and their private utilities as the human individuals in the economy and the associated personal rewards they are each trying to maximize. One could then identify the world utility as the time average of the gross domestic product. ("World utility" per se is not a construction internal to a human economy, but rather something defined from the outside.) To achieve high world utility it is necessary to avoid having the agents work at cross-purposes lest phenomena like liquidity traps or the Tragedy of the Commons (TOC) occur, in which agents' individually pursuing their private utilities lowers world utility [10]. The obvious way to avoid such phenomena is by modifying the agents' utility functions to be "aligned" with the world utility . This can be done via punitive legislation. A real world example of an attempt to do this was the creation of anti-trust regulations designed to prevent monopolistic practices.[1]

We do not insist that the agents in a collective really are "trying" to maximize their

[1] In conventional economics, imposing governmental regulations is viewed as a change in the dynamical laws of the variables constituting the world (e.g., now if you perform the proscribed action A you go to jail, whereas that wasn't the case before). Here instead it is abstracted to be a direct change in the mapping between the state of the world and the agent's utility value, without any change in underlying dynamical laws. (To continue with the example, in this alternative there is no direct introduction of a new variable having to do with some physical jail — rather your utility function is directly changed so that now if you do A, your utility value is smaller than if you do not do A.)

private utilities, in some teleological sense. We only require only that they can be *viewed* that way. This allows us to circumvent the fraught exercise of formulating a definition of what an arbitrary component of some physical system is "trying to do". This is illustrated with another naturally occurring example of a collective: a spin glass. One can take the agents in such a glass to be the individual spins. Each spin's "private utility" is (the negative of) its local Hamiltonian (which is determined only by the states of the other spins to which it is coupled). The "world utility" is (the negative of) the Hamiltonian of the entire system. In this example, both the world utility and all of the private utilities are at (local) maxima at equilibrium. This is what allows us to view the spins *as though* they were agents trying to maximize their private utilities.

In addition to such naturally occurring examples, many current artificial systems can be viewed as collectives. For example, the routers in a terrestrial telecommunications network can be viewed as agents in a collective consisting of the entire network. Real-world routers can reasonably be viewed as "trying to maximize" the quality of service accorded to the traffic that crosses them. Hence they can be taken to be the collective's agents. World utility in this example can then be set to aggregate quality of service of the entire network.

With the advent of ubiquitous cheap computing in the near future, the number of artificial control systems that are collectives should explode. Two obvious examples here are a user's constellation of multiple wearable computers, and "computational clouds" of computationally enabled household devices. If such distributed systems are not to be extremely brittle, then absent centralized communication and control, the individual components of the system will need to be both autonomous and adaptive. Almost by definition, this means that those components will be using statistical and machine learning techniques of some sort to modify their behavior to try to meet a goal, i.e., to maximize their private utility.[2]. Moreover, in both of these examples, there is an obvious choice of world utility: the satisfaction level of the user(s) of the system.

Other more prosaic examples of artificial collectives will be dynamic job migration and/or data migration across heterogenous networks of computers. World utility here will be aggregate satisfaction of the network's users, suitably quantified. As with our previous examples, with centralized control being impractical, for the system to be robust there will have to be adaptive "agents" in the system that can be viewed as trying to maximize associated private utilities. As examples, the individual agents could be the computers in the network, or even the jobs and/or sets of data themselves. Similarly, as soon as associated computational control devices are distributed across such systems, many aspects of the management of supply chain, of electric power grid management, of automobile traffic control and automated control of constellations of deployable autonomous vehicles will constitute collectives.

Another broad class of artificial collectives is essentially every system that will involve copious amounts of nanotechnology where many of the nano-scale components in the system have non-trivial associated computational capabilities. This may include everything from continuously deformable wings to smart paint to nano-scale information storage and retrieval systems.

Finally, consider search algorithms that try to find the value of a high-dimensional variable $\vec{z}$ for which a pre-specified function f has a large value. Examples of such algorithms are

[2]When used for this purpose, such techniques are either explicitly or implicitly related to the field Reinforcement Learning (RL) [15, 25, 24, 29, 2, 7, 11, 16, 21])

gradient ascent, simulated annealing, genetic algorithms, etc. Say we take the final value of f achieved by such an algorithm to be the "world utility" of the entire system's dynamic history. Assuming each individual component of $\vec{z}$ evolves with the "goal" of maximizing that final value of $f(\vec{z})$, we can view each such component as an agent, with private utility given by the final value of f. (Note that the private utility of an agent depends on variables not directly under the agent's control, in general.) In this way any search algorithm can be viewed as a collective. However conventionally such algorithms use very "dumb" agents (e.g., semi-random agents rather than RL-based agents). They also don't consider possible modifications to the underlying system, e.g., to the choice of private utilities, that might result in a better value of final value of f. (The design problem of how best to set private utilities is discussed in the next section.) Constructing search algorithms that use techniques of this nature — intuitively, "agentizing" the individual variables of a search problem by providing them with adaptive intelligence — would provide a search algorithm that is immediately parallelizable. Owing to their use of "smart" variables, such algorithms might also lead to substantially better final values of f than conventional search algorithms.

1.2 The Design of Collectives

The "forward problem" in the science of collectives is how the precise configuration of the system — including in particular the private utilities of the agents — affects the ensuing behavior, and therefore affects the value of the world utility. In light of the examples above however, there is another problem that is at least as rich scientifically, but as a practical matter is of more immediate concern. This is the inverse problem: *How should one initialize/update the private utility functions of the individual agents so that the ensuing behavior of the entire collective achieves large values of the provided world utility?* In particular, since in truly large systems detailed modeling of the system is usually impossible, how can we solve this problem in a way that avoids such modeling? Can we somehow solve it if we leverage only the simple assumption that our agents' learnering algorithms are individually fairly good at what they do?

This design problem is related to work in many other fields, including multi-agent systems (MAS's), computational economics, mechanism design, reinforcement learning, statistical mechanics, computational ecologies, (partially observable) Markov decision processes and game theory. However none of these fields is both applicable in large problems, and directly addresses the *general* inverse problem, rather than a special instance of it. (See [37] for a detailed discussion of the relationship between these fields, involving hundreds of references.)

For example, the subfield of game-theory known as mechanism design might, at first glance, appear to provide us techniques for solving the inverse problem. However mechanism design is almost exclusively concerned with collectives that are at (a suitable refinement of) Nash equilibrium [9, 20, 19]. That means that every agent is assumed to be performing *as well as is theoretically possible*, given the behavior of the rest of the system. In setting private utilities and the like on this basis, mechanism design ignores completely the issue of how to design the system so that each of the agents can achieve a good value of its private utility (given the behavior of the rest of the system). In particular it ignores all statistical issues related to how well the agents can be expected to perform for various candidate private utilities. Such issues become crucial as one moves to large systems, where each agent is implicitly confronted with

a very high-dimensional RL task.

There are many other issues that arise in bounded rational situations that are not considered by mechanism design since they do not arise when there is full rationality. For example, it is often the case that by "stabilizing" the sequence of actions of some agent ρ the other agents, being in a more predictable environment, are able to perform better. Conversely, such enforced stabilization of its actions will often hurt the performance of agent ρ. Mechanism design almost completely ignores the associated issues of how best to trade off the performance of one agent against that of other agents, or more generally of how best to trade off the degree of rationality of one agent against that of another agent. (Indeed, mechanism design does not even possess a broadly applicable model-independent measure of "degree of rationality".)

In addition to these problems, many of the techniques derived in mechanism design cannot be applied in numerous application domains, since those techniques are largely tailored to collectives of human beings. In particular, many of those techniques are tailored to the idiosyncracy of such collectives that their members have hidden variables whose values they "do not want to reveal". This idiosyncracy is reflected in restrictions on the allowed form of the private utilities and the world utility and communication structures among the agents. Indeed, if there were no such restriction, then given the Nash equilibrium presumption of mechanism design, how best to set the private utilities would be a trivial problem: To have the maximum of world utility be a Nash equilibrium, simply set each such private utility to equal the world utility, in a so-called "team game" or an "exact potential game" [8]. To have the analysis be non-trivial, restrictions like those that apply to the private utilities of human beings are needed.

Not only are the techniques of mechanism design not relevant to many domains, because those domains do not have the restrictions assumed in mechanism design, but in addition there are many issues that loom large in such domains about which mechanism design is mute. For example, in computational domains, where the agents are computer programs each controlling a set of certain variables, we often have some freedom to change how the set of all variables being controlled is partitioned among the agents, and even change the number of such agents. Needless to say, with its focus on human agents, mechanism design has little advice to provide on such issues of how best to define the agents in the first place.

Perhaps the most striking illustration of the shortcoming of mechanism design is the fact that it does not allow for run-time adaptive redesign. For real-world bounded rational agents, the initial design of the system necessarily makes assumptions which invariably are at least partially at variance with reality. To address this, one must employ adaptive techniques (e.g., statistical estimation) on the running system to refine one's initial assumptions, and then modify the design accordingly. Yet almost all of mechanism design has no room for addressing such "macro-learning".

There is other previous work that does consider the inverse problem in its broadest sense, and even has each agent explicitly use RL techniques, so that no formal assumption is made in the associated theory that the system is at Nash equilibrium. Despite this use of RL though, in general in that work the private utilities are set as in a team game. So again, there is no concern for how well the agents can discern how best to act to maximize their utilities. Unfortunately, as intimated above (and expounded below), ignoring this issue means that the approach scales extremely poorly to large problems. Intuitively, the difficulty is that each agent will have a hard time discerning the echo of its behavior on its private utility

when the system is large if that private utility is the world utility; each agent has a horrible "signal-to-noise" problem in such a situation.[3]

Intuitively, in designing the private utilities of a collective we want them to be "aligned" with the world utility, in that modifications an agent might make that would increase its private utility also must increase world utility. Fortunately the equivalence class of such private utilities extends well beyond team-game utilities. In particular, it extends to include utilities that have far better "signal-to-noise" properties. By using those utilities one can get far better values of world utility than would otherwise be possible. The mathematical theory for how to generate such alternative private utilities is presented in the next section. The following, last section of this chapter then summarizes many experiments that demonstrate that by using those alternative private utilities one can improve performance by up to orders of magnitude, and that the gain in performance grows as the system gets larger.

2 The Mathematics of Designing Collectives

In this chapter attention is restricted to collectives in which the individual agents are prefixed, being the players in multi-stage non-cooperative games, with their moves at any single stage in no *a priori* way restricted by their moves at other times or by the moves of the other players. Some techniques for the design of the private utilities in such games are known as the "COllective INtelligence (COIN)" framework.[38] This section presents some of the mathematics necessary to understand that framework. It should be emphasized however that the full mathematics of how to design collectives extends significantly beyond what is needed to address such games. [4]

The restricted version of that full mathematics needed to present the COIN framework starts with an arbitrary vector space Z whose elements ζ give the joint move of all players in the collective in some stage. We wish to search for the ζ that maximizes the provided world utility $G(\zeta)$. In addition to G we are concerned with private 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 some arbitrary 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:

$$\epsilon_{\eta,U}(\zeta) \equiv \int d\mu_{\zeta_{\hat{\eta}}}(\zeta')\Theta[U(\zeta) - U(\zeta')] \; , \tag{1}$$

where the Heaviside function Θ is defined to equal 1 when its argument is greater than or

[3] To help see this, consider the example of a collective provided by the human economy. A team game in that example would mean that every human gets US GDP as its reward signal, and tries to discern how best to act to maximize that reward signal. At the risk of understatement, this would provide the individual members of the economy with a difficult reinforcement learning task.

[4] That 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 across all space and time are assigned to players), as well as modification of the players' information sets (i.e., modification of inter-player communication). See [33].

equal to 0, and to equal 0 otherwise, and where the subscript on the (normalized) measure $d\mu$ indicates it is restricted to ζ' sharing the same non-η components as ζ.[5] Intelligence value 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 private utility functions of the players. Indicating that value of the **global coordinate** by s, our analysis revolves around the following **central equation** for $P(G \mid s)$, which follows from Bayes' theorem:

$$P(G \mid s) = \int d\vec{\epsilon}_G P(G \mid \vec{\epsilon}_G, s) \int d\vec{\epsilon}_g P(\vec{\epsilon}_G \mid \vec{\epsilon}_g, s) P(\vec{\epsilon}_g \mid s) \; , \tag{2}$$

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

Note that $\epsilon_{\eta, g_\eta}(\zeta) = 1$ means that player η is fully rational at ζ, in that its move maximizes the value of its utility, given the moves of the players. In other words, a point ζ where $\epsilon_{\eta, g_\eta}(\zeta) = 1$ for all players η is one that meets the definition of a game-theory Nash equilibrium.[6] 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). So if we can get these two vectors to be identical, then if the agents do well enough at maximizing their private utilities we are assured we will be near a local maximum of G.

To formalize this, consider our decomposition of $P(G \mid s)$. 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 (private utility 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 private utility functions have high "signal-to-noise" arises in the third term. It is in the second term that the requirement that the private utility functions be "aligned with G" arises. In this chapter we concentrate on these two terms, and show how to simultaneously set them to have the desired form. [7]

Details of the stochastic environment in which the collective operates, together with details of the learning algorithms of the players, are all bundled into the distribution $P(\zeta)$ which

[5] The measure must reflect the type of system at hand, e.g., whether Z is countable or not, and if not, what coordinate system is being used. Other than that, any convenient choice of measure may be used and the theorems will still hold.

[6] See [9]. Note that consideration of points ζ at which *not* all intelligences equal 1 provides the basis for a model-independent formalization of bounded rationality game theory. This formalization contains variants of many of the theorems of conventional full-rationality game theory. See [32].

[7] Search algorithms that do not involve game theory (e.g., simulated annealing) can be viewed as addressing how to have term 1 have the desired form. They do this by trying to ensure that the particular local maximum they find of the function they are searching has a high value of that function. This is the essence of why such algorithms "trade off exploration and exploitation". One can combine such term-1-based techniques with the techniques presented in this paper. Intuitively, this amounts to "wrapping" a system using the private utilities derived below in an outer loop that trades off exploration and exploitation. The resultant hybrid algorithm, addressing all three terms, outperforms simulated annealing by over two orders of magnitude[34].

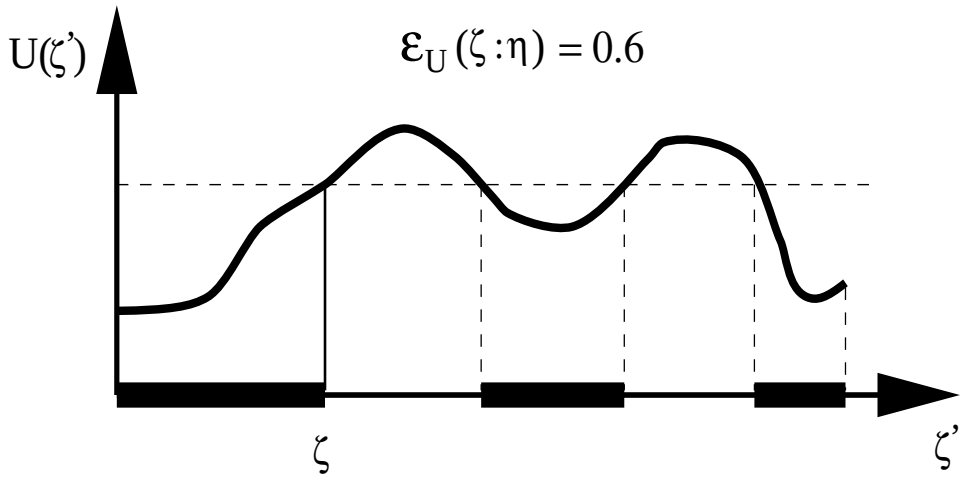

Figure 1: Intelligence of agent η at state ζ for utility U is the actual joint move at hand. The x-axis shows agent η's alternative possible moves (all states ζ' having ζ's values for the moves of all players other than η.). The thick sections of the x-axis show the alternative moves that η could have made that would have given η a worse value of the utility U. The fraction of the full set of η's possible moves that lies in those thick sections (which is 0.6 in this example) is the intelligence of agent η at ζ for utility U, denoted by $\epsilon_{\eta,U}(\zeta)$.

underlies the distributions appearing in Equation 2. Note though that *independent of these considerations*, our desired form for the second term in Equation 2 is assured if we have chosen private utilities such that $\vec{\epsilon}_g$ equals $\vec{\epsilon}_G$ exactly for all ζ. Such a system is said to be **factored**. In game-theory parlance, 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 (an issue rarely considered in the game theory / mechanism design literature).

As a trivial example, any "team game" in which all the private utility functions equal G is factored [8, 17]. However team games often have very poor forms for term 3 in Equation 2, forms which get progressively worse as the size of the collective grows. This is because for such private utility functions each player η will usually confront a very poor "signal-to-noise" ratio in trying to discern how its actions affect its utility $g_\eta = G$, since so many other player's actions also affect G and therefore dilute η's effect on its own private utility function.

We now focus on algorithms based on private utility functions $\{g_\eta\}$ that optimize the signal/noise ratio reflected in the third term, subject to the requirement that the system be factored. To understand how these algorithms work, say we are given an arbitrary function $f(\zeta_\eta)$ over player η's moves, two such moves ${\zeta_\eta}^1$ and ${\zeta_\eta}^2$, a utility U, a value s of the global coordinate, and a move by all players other than η, $\zeta_{\hat{\eta}}$. Define the associated **learnability** by

$$\Lambda_f(U; \zeta_{\hat{\eta}}, s, {\zeta_\eta}^1, {\zeta_\eta}^2) \equiv \sqrt{\frac{[E(U; \zeta_{\hat{\eta}}, {\zeta_\eta}^1) - E(U; \zeta_{\hat{\eta}}, {\zeta_\eta}^2)]^2}{\int d\zeta_\eta [f(\zeta_\eta) Var(U; \zeta_{\hat{\eta}}, \zeta_\eta)]}} \; . \tag{3}$$

The expectation values in the numerator are formed by averaging over the training set of the learning algorithm used by agent η, n_η. Those two averages are evaluated according to the two distributions $P(U|n_\eta)P(n_\eta|\zeta_{\hat{\eta}}, {\zeta_\eta}^1)$ and $P(U|n_\eta)P(n_\eta|\zeta_{\hat{\eta}}, {\zeta_\eta}^2)$, respectively. (That is the

meaning of the semicolon notation.) Similarly the variance being averaged in the denominator is over n_η according to the distribution $P(U|n_\eta)P(n_\eta|\zeta_{\hat{\eta}}, \zeta_\eta)$.

The denominator in Equation 3 reflects how sensitive $U(\zeta)$ is to changing $\zeta_{\hat{\eta}}$. In contrast, the numerator reflects how sensitive $U(\zeta)$ is to changing ζ_η. So the greater the learnability of a private utility 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, so long as it does not come at the expense of decreasing the signal, increasing the signal-to-noise ratio specified in the learnability will make it easier for η to achieve a large value of its intelligence. This can be established formally: if appropriately scaled, g'_η will result in better expected intelligence for agent η than will g_η whenever $\Lambda_f(g'_\eta; \zeta_{\hat{\eta}}, s, {\zeta_\eta}^1, {\zeta_\eta}^2) > \Lambda_f(g_\eta; \zeta_{\hat{\eta}}, s, {\zeta_\eta}^1, {\zeta_\eta}^2)$ for all pairs of moves ${\zeta_\eta}^1, {\zeta_\eta}^2$[33]. [8]

It is possible to solve for the set of all private utilities that are factored with respect to a particular world utility. Unfortunately, in general it is not possible for a collective both to be factored and to have infinite learnability for all of its players. However consider **difference** utilities, which are of the form

$$U(\zeta) = \beta[G(\zeta) - \Gamma(\zeta_{\hat{\eta}})] \tag{4}$$

Any difference utility is factored [33]. In addition, under usually benign approximations, $\Lambda_f(U; \zeta_{\hat{\eta}}, s, {\zeta_\eta}^1, {\zeta_\eta}^2)$ is maximized over the of difference utilities for all pairs ${\zeta_\eta}^1, {\zeta_\eta}^2$ by choosing

$$\Gamma(\zeta_{\hat{\eta}}) = E_f(G(\zeta) \mid \zeta_{\hat{\eta}}, s) \ , \tag{5}$$

up to an overall additive constant, where the expectation value is over ζ_η. 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 each player η uses an appropriately rescaled version of the associated AU as its private utility function, then we have ensured good form for both terms 2 and 3 in Equation 2.

Using AU in practice is sometimes difficult, due to the need to evaluate the expectation value. Fortunately there are other utility functions that, while being easier to evaluate than AU, still are both factored and possess superior learnability to the team game utility, $g_\eta = G$. One such private utility function is the **Wonderful Life** Utility (WLU). The WLU for player η is parameterized by a pre-fixed **clamping parameter** CL_η chosen from among η's possible moves:

$$WLU_\eta \equiv G(\zeta) - G(\zeta_{\hat{\eta}}, CL_\eta) \ . \tag{6}$$

WLU is factored no matter what the choice of clamping parameter. Furthermore, while not matching the high learnability of AU, WLU usually has far better learnability than does a team game, and therefore (when appropriately scaled) results in better expected intelligence [28, 39, 37, 41].

Figure 2 provides an example of clamping. As in that example, in many circumstances there is a particular choice of clamping parameter for player η that is a "null" move for that

[8] In many RL algorithms, changing the scale of the utility is exactly equivalent to changing a "temperature" parameter of the algorithm. Such temperatures have to usually be set via a search process. The result presented here establishes that so long as g'_η has higher learnability than does g_η, the expected intelligence of g'_η at the associated optimal temperature will be higher than that of g_η at its optimal temperature.

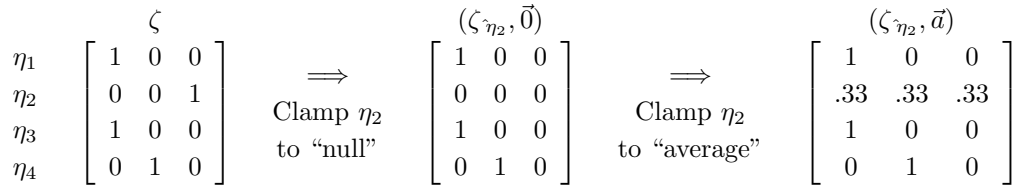

Figure 2: This example shows the impact of the clamping operation on the joint state of a four-player system where each player has three possible moves, each such move represented by a three-dimensional unary vector. The first matrix represents the joint state of the system ζ where player 1 has selected action 1, player 2 has selected action 3, player 3 has selected action 1 and player 4 has selected move 2. The second matrix displays the effect of clamping player 2's action to the "null" vector (i.e., replacing ζ_{η_2} with $\vec{0}$). The third matrix shows the effect of instead clamping player 2's move to the "average" action vector $\vec{a} = \{.33, .33, .33\}$, which amounts to replacing that player's move with the "illegal" move of fractionally taking each possible move ($\zeta_{\eta_2} = \vec{a}$).

player, equivalent to removing that player from the system. (Hence the name of this private utility function — cf. the Frank Capra movie.) For such a clamping parameter assigning the associated WLU to η as its private utility function is closely related to the economics technique of "endogenizing a player's externalities", for example with the Groves mechanism [18, 19, 9].

However it is usually the case that using WLU with a clamping parameter that is as close as possible to the expected move defining AU results in far higher learnability than does clamping to the null move. Such a WLU is roughly akin to a mean-field approximation to AU.[9] For example, in Fig. 2, if the probabilities of player 2 making each of its possible moves was 1/3, then one would expect that a clamping parameter of $\vec{a}$ would be close to optimal. Accordingly, in practice use of such an alternative WLU derived as a "mean-field approximation" to AU almost always results in far better values of G than does the "endogenizing" WLU.

Intuitively, collectives having factored and highly learnable private utilities like AU and WLU can be viewed as akin to well-run human companies. G is the "bottom line" of the company, the players η are identified with the employees of that company, and the associated g_η given by the employees' performance-based compensation packages. For example, for a "factored company", each employee's compensation package contains incentives designed such that the better the bottom line of the corporation, the greater the employee's compensation. As an example, the CEO of a company wishing to have the private utilities of the employees be factored with G may give stock options to the employees. The net effect of this action is to ensure that what is good for the employee is also good for the company. In addition, if the compensation packages are "highly learnable", the employees will have a relatively easy time discerning the relationship between their behavior and their compensation. In such a case the employees will both have the incentive to help the company and be able to determine how best to do so. Note that in practice, providing stock options is usually more effective

[9] Formally, our approximation is exact only if the expected value of G equals G evaluated at the expected joint move (both expectations being conditioned on given moves by all players other than η). In general though, for relatively smooth G, we would expect such a mean-field approximation to AU, to give good results, even if the approximation does not hold exactly.

in small companies than in large ones. This makes perfect sense in terms of the formalism summarized above, since such options generally have higher learnability in small companies than they do in large companies, in which each employee has a hard time seeing how his/her moves affect the company's stock price.

3 Tests of the Mathematics

As a test of the preceding mathematics, in some of our previous work we used the WLU for distributed control of network packet routing [39]. Conventional approaches to packet routing have each router run a shortest path algorithm (SPA), i.e., each router routes its packets in the way that it expects will get those packets to their destinations most quickly. Unlike with a WLU-based collective, with SPA-based routing the routers have no concern for the possible deleterious side-effects of their routing decisions on the global goal (e.g., they have no concern for whether they induce bottlenecks). We ran simulations that demonstrated that a WLU-based collective has substantially better throughputs than does the best possible SPA-based system [39], even though that SPA-based system has information denied the agents in the WLU-based collective.

In related work we have shown that use of the WLU automatically avoids the infamous Braess' paradox, in which adding new links can actually decrease throughput — a situation that readily ensnares SPA's.[28, 36]

In yet other work we have applied the WLU to the problem of controlling communication across a constellation of satellites so as minimize the importance-weighted loss of scientific data flowing across that constellation.[35] Due to the novelty of this problem domain, we first had to design a "baseline" distributed control algorithm, one that involves no learning. To minimize the number of confounding distinctions between that baseline algorithm and the collective-based algorithm we investigated, we had that collective "run on top" of the baseline algorithm. The action of each agent in the collective was the determination of fictitious "ghost traffic" that is presented to the baseline algorithm, thereby (hopefully) inducing that baseline algorithm to achieve an even better value of the world utility. (Note that this idea can be applied with most any baseline algorithm and most any distributed RL algorithm.) Again, we achieved a significant increase in performance, in this case relative to the baseline algorithm.

We have also successfully applied the COIN techniques to problems that are explicitly cast as search. These include setting the states of the spins in a spin glass to minimize energy; the conventional bin-packing problem of computer science, and a model of human agents connected in a small-world network who have to synchronize their purchase decisions. We have also successfully applied COIN techniques to the problem of coordinativing a set of autonomous rovers so as to maximize the importance-weighted value of a set of locations they visit.[27].

Finally, it is worth going into some detail our investigations of variants of congestion games [40, 38, 41, 38], in particular of a more challenging variant of Arthur's El Farol bar attendance problem [1], sometimes also known as the "minority game" [6]. In this problem the individual processes making up the collective are explicitly viewed as 'players' involved in a non-cooperative game. Each player has to determine which night in the week to attend a bar. The problem is set up so that if either too few people attend (boring evening) or too

many people attend (crowded evening), the total enjoyment of the attending players drops. Our goal is to design the private utility functions of the players so that the total enjoyment across all nights is maximized. In this previous work we showed that use of the WLU can result in performance *orders of magnitude* superior to that of team game utilities.

Arthur's bar problem [1] can be viewed as a problem in designing collectives. Loosely speaking, in this problem at each time step each player η decides whether to attend a bar by predicting, based on its previous experience, whether the bar will be too crowded to be "rewarding" at that time, as quantified by a utility function G. The selfish nature of the players frustrates the global goal of maximizing G. This is because if most players think the attendance will be low (and therefore choose to attend), the attendance will actually be high, and vice-versa.

We variants of the bar problem we investigated were all of the following type: There are N players, each picking one out of seven moves every week. Each variant of the game is parameterized by $\ell \in \{1, 2, 3, 4, 5, 6\}$. In a given variant, each move of an agent corresponds to attending the bar on some particular subset of ℓ out of the seven nights of the current week (i.e., given ℓ, each possible move is an 'attendance profile' vertex of the 7-dimensional unit hypercube having ℓ 1's). In each week every player chooses a move. Then the associated private utility values for each player are communicated to that player, and the process is repeated. For simplicity, for each ℓ we chose the seven possible attendance profiles so that if the moves are selected randomly uniformly, the expected resultant attendance profile across all seven nights is also uniform. (For example, or $\ell = 2$, those profiles are (1, 1, 0, 0, 0, 0, 0), (0, 1, 1, 0, 0, 0, 0), etc.)

More formally, the world utility in any particular week is:

$$G(\zeta) \equiv \sum_{k=1}^{7} \phi(x_k(\zeta)) \; , \tag{7}$$

where $x_k(\zeta)$ is the total attendance on night k; ζ_η is η's move in that week; $\phi(y) \equiv y \exp(-y/c)$; and c is a real-valued parameter. Our choice of $\phi(.)$ means that when either too few or too many players attend some night in some week world utility G is low.

Since we wished to concentrate on the effects of the utilities rather than on the RL algorithms that use them, we used (very) simple RL algorithms.[10] We would expect that even marginally more sophisticated RL algorithms would give better performance. In our algorithm each player η had a 7-dimensional vector giving its estimates of the utility it would receive for taking each possible move. At the beginning of each week, each η picked the night to attend randomly, using a Boltzmann distribution over the seven components of η's estimated utilities vector. For simplicity, temperature did not decay in time. However to reflect the fact that each player operated in a non-stationary environment, utility estimates were formed using exponentially aged data: in any week t, the estimate η makes for the utility for attending night i was a weighted average of all the utilities it has previously received when it attended that night, with the weights given by an exponential function of how long ago each such utility was. To form the players' initial training set, we had an initial period in which all moves by all players were chosen uniformly randomly, with no learning.

[10] On the other hand, to use algorithms so patently deficient that they have never even been considered in the RL community — like the algorithms used in most of the bar problem literature — would seriously interfere with our ability to interpret our experiments.

In these experiments we found once again that use of highly learnable factored private utilities resulted in vastly better performance than use of team game private utilities. Also as usual, we found that the gain in performance grew as the problem grew, reaching orders of magnitude once the system grew to consist of thousands of agents. We always found that AU performed at least as well as WLU with clamping to 0, which is essentially identical to the economics technique of "endogenizing externalities". In addition though, for some choices of ℓ, we found that AU performed substantially better than did this alternative, as would be expected based on the formalism presented above.

Finally, the central equation makes numerous other predictions that preliminary experiments seem to bear out. Some of these predictions concern ways to modify the behavior of the collective to try to optimize term 1 as well as terms 2 and 3. (The work in [35] can be viewed as an initial investigation of this issue.) Other predictions are on how to modify a factored private utility so that it is *not* perfectly factored any more, but has undergone such a large gain in learnability that (as quantified in the central equation) overall performance improves.[31] It is worth emphasizing that such beneficial modifications to private utilities are prohibited by the starting premise of the field of mechanism design, that the private utilities must exhibit "incentive compatibility".

4 Conclusion

A collective is any multi-agent system in which each agent adaptively tries to maximize its own un private utility, while at the same time there is an overall world utility rating the behavior of the entire system. Collectives are quite common in the natural world, the canonical example being any human organization. In addition, as computing becomes ubiquitous in artificial systems, the number of such systems that constitute collectives will explode.

Associated with any collective is a design problem, of how to configure the system — and in particular how to set the private utilities of the agents — to optimize the world utility. This paper cursorily synopsizes some of the mathematical theory underpinning this design problem. That theory has now been tested in many different experiments. As summarized above, those tests have the theory have clearly validated it, often resulting in performance up to orders of magnitude superior to traditional techniques from the fields of multi-agent systems and economics/mechanism design. Intuitively, that superiority lies in the fact that these alternative approaches completely ignore the issue of how an agent's ability to maximize a candidate private utility will vary with changes in that private utility. This issue is especially crucial in large systems, in which each agent will face an extremely difficult "signal-to-noise" term in discerning the effects of its actions on its utility unless that utility is carefully crafted.

References

[1] W. B. Arthur. Complexity in economic theory: Inductive reasoning and bounded rationality. *The American Economic Review*, 84(2):406–411, May 1994.

[2] C. Boutilier. Multiagent systems: Challenges and opportunities for decision theoretic planning. *AI Magazine*, 20:35–43, winter 1999.

[3] C. Boutilier, Y. Shoham, and M. P. Wellman. Editorial: Economic principles of multi-agent systems. *Artificial Intelligence Journal*, 94:1–6, 1997.

[4] J. M. Bradshaw, editor. *Software Agents*. MIT Press, 1997.

[5] G. Caldarelli, M. Marsili, and Y. C. Zhang. A prototype model of stock exchange. *Europhysics Letters*, 40:479–484, 1997.

[6] D. Challet and Y. C. Zhang. On the minority game: Analytical and numerical studies. *Physica A*, 256:514, 1998.

[7] 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.

[8] 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.

[9] D. Fudenberg and J. Tirole. *Game Theory*. MIT Press, Cambridge, MA, 1991.

[10] G. Hardin. The tragedy of the commons. *Science*, 162:1243–1248, 1968.

[11] J. Hu and M. P. Wellman. Multiagent reinforcement learning: Theoretical framework and an algorithm. In *Proceedings of the Fifteenth International Conference on Machine Learning*, pages 242–250, June 1998.

[12] B. A. Huberman and T. Hogg. The behavior of computational ecologies. In *The Ecology of Computation*, pages 77–115. North-Holland, 1988.

[13] N. R. Jennings, K. Sycara, and M. Wooldridge. A roadmap of agent research and development. *Autonomous Agents and Multi-Agent Systems*, 1:7–38, 1998.

[14] 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.

[15] L. P. Kaelbing, M. L. Littman, and A. W. Moore. Reinforcement learning: A survey. *Journal of Artificial Intelligence Research*, 4:237–285, 1996.

[16] M. L. Littman. Markov games as a framework for multi-agent reinforcement learning. In *Proceedings of the 11th International Conference on Machine Learning*, pages 157–163, 1994.

[17] D. Monderer and L. S. Sharpley. Potential games. *Games and Economic Behavior*, 14:124–143, 1996.

[18] W. Nicholson. *Microeconomic Theory*. The Dryden Press, seventh edition, 1998.

[19] N. Nisan and A. Ronen. Algorithmic mechanism design. *Games and Economic Behavior*, 35:166–196, 2001.

[20] D. C. Parkes. *Iterative Combinatorial Auctions: Theory and Practice*. PhD thesis, University of Pennsylvania, 2001.

[21] T. Sandholm and R. Crites. Multiagent reinforcement learning in the iterated prisoner's dilemma. *Biosystems*, 37:147–166, 1995.

[22] T. Sandholm, K. Larson, M. Anderson, O. Shehory, and F. Tohme. Anytime coalition structure generation with worst case guarantees. In *Proceedings of the Fifteenth National Conference on Artificial Intelligence*, pages 46–53, 1998.

[23] S. Sen. *Multi-Agent Learning: Papers from the 1997 AAAI Workshop (Technical Report WS-97-03.* AAAI Press, Menlo Park, CA, 1997.

[24] R. S. Sutton. Learning to predict by the methods of temporal differences. *Machine Learning*, 3:9–44, 1988.

[25] R. S. Sutton and A. G. Barto. *Reinforcement Learning: An Introduction.* MIT Press, Cambridge, MA, 1998.

[26] K. Sycara. Multiagent systems. *AI Magazine*, 19(2):79–92, 1998.

[27] K. Tumer, A. Agogino, and D. Wolpert. Learning sequences of actions in collectives of autonomous agents. In *Proceedings of the First International Joint Conference on Autonomous Agents and Multi-Agent Systems*, Bologna, Italy, July 2002.

[28] 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.

[29] C. Watkins and P. Dayan. Q-learning. *Machine Learning*, 8(3/4):279–292, 1992.

[30] M. P. Wellman. A market-oriented programming environment and its application to distributed multicommodity flow problems. In *Journal of Artificial Intelligence Research*, 1993.

[31] D. Wolpert and J. Lawson. Designing agent collectives for systems with markovian dynamics. In *Proceedings of the First International Joint Conference on Autonomous Agents and Multi-Agent Systems*, Bologna, Italy, July 2002.

[32] D. H. Wolpert. Bounded-rationality game theory. pre-print, 2001.

[33] D. H. Wolpert. Theory of design of collectives. pre-print, 2002.

[34] D. H. Wolpert, E. Bandari, and K. Tumer. Improving simulated annealing by recasting it as a non-cooperative game. 2001. submitted.

[35] D. H. Wolpert, J. Sill, and K. Tumer. Reinforcement learning in distributed domains: Beyond team games. In *Proceedings of the Seventeenth International Joint Conference on Artificial Intelligence*, pages 819–824, Seattle, WA, 2001.

[36] D. H. Wolpert and K. Tumer. Collective intelligence, data routing and braess' paradox. *Journal of Artificial Intelligence Research*, 1995. to appear.

[37] D. H. Wolpert and K. Tumer. An Introduction to Collective Intelligence. Technical Report NASA-ARC-IC-99-63, NASA Ames Research Center, 1999. URL:`http://ic.arc.nasa.gov/ic/projects/coin_pubs.html`. To appear in Handbook of Agent Technology, Ed. J. M. Bradshaw, AAAI/MIT Press.

[38] D. H. Wolpert and K. Tumer. Optimal payoff functions for members of collectives. *Advances in Complex Systems*, 4(2/3):265–279, 2001.

[39] 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.

[40] D. H. Wolpert, K. Wheeler, and K. Tumer. General principles of learning-based multi-agent systems. In *Proceedings of the Third International Conference of Autonomous Agents*, pages 77–83, 1999.

[41] D. H. Wolpert, K. Wheeler, and K. Tumer. Collective intelligence for control of distributed dynamical systems. *Europhysics Letters*, 49(6), March 2000.

[42] Y. C. Zhang. Modeling market mechanism with evolutionary games. *Europhysics Letters*, March/April 1998.