Overcoming Communication Restrictions in Collectives

Kagan Tumer
Computational Sciences Division
NASA Ames Research Center
Mailstop 269-4
Moffett Field, CA 94035
E-mail: ktumer@mail.arc.nasa.gov

Adrian K. Agogino
University of California, Santa Cruz
NASA Ames Research Center
Mailstop 269-3
Moffett Field, CA 94035
E-mail: adrian@email.arc.nasa.gov

***Abstract*— The performance of distributed systems generally depend on the actions and interactions of a large number of independent components (e.g., agents, neurons). Such "collectives" are often subject to communication restrictions, making it difficult for the components to coordinate their actions to provide good system level performance. In this article we address that coordination problem and derive four agent utility functions that make different tradeoffs between alignedness between agent and system utilities and the signal-to-noise each agent encounters. The results show that these utility functions outperform both traditional methods and previous collective-based methods by up to 75% in systems with communication restrictions.**

I. Introduction

[1] Control and coordination in a large distributed system that needs to achieve a predetermined task is a challenging area of research. Many methods exist for coordinating the actions of the components (e.g., agents, neurons) of such a system when those components can fully communicate with one another [6], [15], [21]. In this work we focus on a solution to this coordination problem based on "collectives" [17], [21]. A collective is a large distributed system of interacting agents where there is a well-defined "world utility" function rating the performance of the full system, and where each agent is only concerned with maximizing its own "private utility" function [21]. However, in many problems, the presence of communication restrictions significantly complicates the coordination problem [4], [8], [13]. Examples of such problems include controlling collections of rovers or constellations of satellites, and coordinating data routing across a network (because of such examples, we will refer to the components as "agents" in this article). In each of those cases, an agent may only be able to directly communicate with a small number of other agents. In addition, even if there are indirect methods for sharing information (e.g., team formation), they may be costly and an agent may be unwilling to share, if doing so would hurt its private utility. In all of these problems, the system designer faces the difficult task of providing the agents with a private utility that:

1) allows agents to work towards the common goal and not against one another, i.e., the agents' private utility functions are aligned with the "world utility function"; and
2) does not require access to global information available through a broad communication network, i.e., agents can determine which actions are beneficial to their private utilities with the limited information at their disposal.

These issues are at odds with each other and in fact in many cases it will be impossible for the agents to achieve high values of a private utilities which is "aligned" with the world utility.[2] In addition even if the world utility, computed with global information, can be broadcast to all the agents, agents may not be able to effectively use this information to select actions that will be useful to them and to the overall system. In fact many obvious methods of combining local information with the world utility can actually cause reduced performance as communication increases (Figure 1). This example shows the behavior of a system (described in detail in Section IV) where the world utility is plotted with respect to the percentage of agents with which an agent can communicate. Note that in some states of the system (e.g., low communication levels), increasing the amount of information to which agents have access has deleterious effects on the performance of the system. We will discuss the reasons for this paradox and show how some problems stemming from communication restrictions can be overcome by providing agents with carefully crafted private utility functions.

The first step in creating a distributed system that can effectively maximize world utility is to ensure that the agents work together. If the agents are not designed to work well with each other, they may not learn their task properly, may interfere with each other's ability to contribute to the world utility, or simply perform useless repetitive work. Hand tailoring the agents' private utility functions may offer a solution, but generally, such systems: (i) have to be laboriously modeled; (ii) provide "brittle" global performance; (iii) are not "adaptive" to changing environments; and (iv) generally do not scale well.

To sidestep these problems, yet address the design requirements listed above (i.e., utility "alignedness" and "learnabil-

[1] Appears in: *Proceedings of IJCNN*, Budapest, Hungary, 2004

[2] By "aligned" we mean that actions that improve the private utility of an agent will also improve the world utility. We will formalize this concept in Section II.

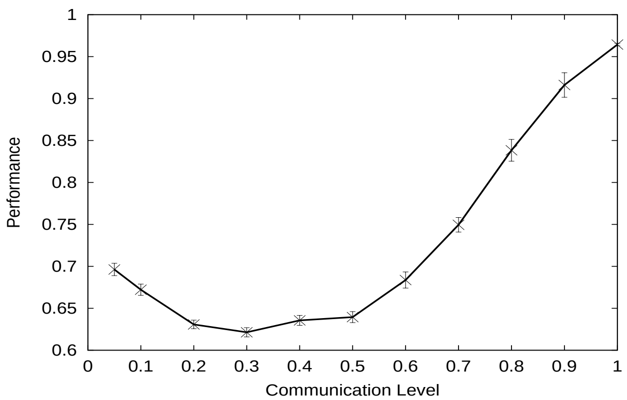

Fig. 1. Sample performance vs. communication level in a system (details in Section IV). Increasing the amount of information at low levels of communication can hurt performance rather than improve it. Only when the communication level reaches a certain threshold does the system performance go up with increasing amount of communication between system components.

ity") one can use the framework of collectives [3] [18], [21]. Given this framework, the crucial design problem becomes: Assuming the individual agents are able to maximize their own utility functions (e.g., through reinforcement learning or evolving neural networks), what set of private utilities for the individual agents will, when pursued by those agents, result in high world utility? The collectives framework has been successfully applied to multiple domains including packet routing over a data network [18], congestion games [21], multiple-resource job scheduling over a heterogeneous computational grid [16], and the coordination of multi-rovers in learning sequences of actions [15].

In this article, we extend the question of how to design the agents' private utilities given that centralized communication is not possible. Though this question has not been directly addressed, there is a large body of work on systems with low levels of communication. Issues such as agent communication languages and physical implementation of communication have received particular attention [7], [14]. At a higher level Pynadath and Tambe have formalized many aspects of agent communications [13], including observability and explicit communication. For multi-agent Markov decision processes, Xule et al. dealt with the problem of partially hidden states of other agents [22]. Furthermore, many researchers have demonstrated that often little communication is needed to coordinate agents [3], and that in many cases local communication is sufficient [8]. However these observations are only true in certain specific domains. In this work, we further explore this tradeoff of global coordination and local information.

In this article, we show how communication restrictions in a system can be overcome by modifying the agents' utilities. Based on the work on collectives, we derive four different agent utility functions that offer different levels of alignedness and learnability for the agents' private utility functions. Furthermore, those utilities differ in whether they allow for global broadcasts of the world utility (in some domains, even though the agents will not be able to engage in realtime agent to agent communication, some global information can be broadcast at various intervals). In Section II, we summarize the theory of collectives that is needed for this article. In Section III, we describe the problem domain and derive the collective-based solution to this problem. In Section IV, we present and discuss the simulation results.

[3]The design of a collective problem is related to work in many fields beyond multiagent systems, including mechanism design, reinforcement learning for adaptive control, computational ecologies, and game theory. See [17] for a detailed survey of collectives and related fields.

II. Background: Collectives

In this section, we summarize the theory of collectives necessary to derive the agent utility functions used in this article. Let Z be an arbitrary vector space whose elements z give the joint move of all agents in the system (i.e., z specifies the full state of the system). The **world utility** $G(z)$, is a function of the full state z, and the problem we face is to find the z that maximizes $G(z)$. In addition to G, each agent η has a **private utility function** g_η. The agents' goals are to optimize their individual private functions, even though, we, as system designers are only concerned with the value of the world utility G. We will denote the state of agent η by z_η, and the state of all *other than* η, by $z_{-\eta}$. In this work we take z, z_η, and $z_{-\eta}$ to have the same dimensionality (e.g., for z_η all elements of z that are not dependent of η are replaced with zeros), resulting in the notation: $z = z_\eta + z_{-\eta}$.

A. Factoredness and Learnability

For high values of G to be achieved, the private utility functions need to have two properties, which we will call **factoredness** and **learnability**. First we want the private utility functions of each agent to be aligned with respect to G, intuitively meaning that an action taken by an agent that improves its private utility also improves the world utility. Specifically, for any two states z and z' which differ only on agent η's state, an action by agent η that increases g_η will also increase G. Formally a utility g_η is factored with G when:

$$g_\eta(z) > g_\eta(z') \leftrightarrow G(z) > G(z') \\ \forall z, z' \;\; s.t. \;\; z_{-\eta} = z'_{-\eta} \quad .$$

In game theory language, the Nash equilibria of a factored system are local maxima of G. In addition to this desirable equilibrium behavior, factored systems also automatically provide appropriate off-equilibrium incentives to the agents (an issue rarely considered in the game theory / mechanism design literature).

Second, we want the agents' private utility functions to have high **learnability**, intuitively meaning that an agent's utility should be sensitive to its own actions and insensitive to actions of others. As a trivial example, any system in which all the private utility functions equal G is factored [6]. However such systems often suffer from low signal-to-noise, a problem that get progressively worse as the size of the system grows. This problem happens since for large systems where G sensitively depends on all components of the system, each agent may experience difficulty discerning the effects of its actions on G. As a consequence, each η may have difficulty achieving high g_η. This signal-to-noise effect, called **learnability** is the second property that is crucial in the design of the agents' private utility functions. Formally we can quantify the learnability of a utility g_η by:

$$\lambda_{\eta,g_\eta}(z) \equiv \frac{\|\vec{\nabla}_{z_\eta} g_\eta(z)\|}{\|\vec{\nabla}_{z_{-\eta}} g_\eta(z)\|} \ . \tag{1}$$

So at a given state z, the higher the learnability, the more $g_\eta(z)$ depends on the move of agent η, i.e., the better the associated signal-to-noise ratio for η. Intuitively then, higher learnability means it is easier for η to achieve a large values of its utility.

B. Difference Utilities

Consider **difference** utilities, which are of the form:

$$DU\eta \equiv G(z) - G(z - z_\eta + v_\eta) \tag{2}$$

where v_η is a constant vector. In the second term of DU, all states depending on η are replaced by a constant, creating a virtual state. Difference utilities are factored no matter the choice of v_η precisely because the second term does not depend on η's state [21]. Furthermore, they usually have far better learnability than does setting g_η to G because the second term of DU removes a lot of the effect of other agents (i.e., noise) from η's utility. In this work we set v_η to the "null" vector, (e.g., $v_\eta = \vec{0}$). Note, that when the null state is used, DU is closely related to the economics technique of "endogenizing a player's (agent's) externalities" [12]. Indeed, DU has conceptual similarities to Vickrey tolls [19], and Groves' mechanism [10], though the Groves mechanism results in a team game.

Intuitively, one can look at DU from the perspective of a human company, with G, the "bottom line" of the company, the agents η, the employees of that company, and the associated g_η, the employees' performance-based compensation packages. For a "factored company", each employee's compensation package contains incentives designed such that the better the bottom line of the company, the greater the employee's compensation. For example the board 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 have "high learnability", 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 generally more effective in small companies than in large ones. This makes perfect sense in terms of the formalism, 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.

TABLE I
COMPARISON OF UTILITY TRADEOFFS

Utility	Factoredness	Learnability	Required Communication
DU	Full	High	Global
BTU	Full	Low	Broadcast/Local
TTU	Partial (low)	High	Local
BEU	Full	Low	Broadcast/Local
EEU	Partial (high)	High	Local

C. Communication Restrictions

In many real world problems the computation of the difference utility requires sufficient communication among the agents to allow the agents to infer the value of the state of the entire system. In some specific domains, using difference utilities results in many elements of the system state to cancel out, allowing the agents to compute DU without knowing the full state. However in general, an agent may not have sufficient communication to compute DU, and needs to approximate under the constraints of communication restrictions.

Mathematically we represent the communication restrictions for an agent η as elements of the system state that are not observable. We can can decompose the state z into a component observable by agent η, z^{o_η}, and a component hidden from agent η, z^{h_η} (note $z = z^{o_\eta} + z^{h_\eta}$). In this paper we will define the communication level for agent η as:

$$B_\eta = \frac{\int_{z^{o_\eta}} dz'}{\int_z dz'} . \tag{3}$$

For a problem with countable state elements, B_η reduces to the number of observable elements in the state divided by the total number of elements in the state. Note that B is always in the range $[0.0, 1.0]$.

If the DU for agent η depends on any component of z^{h_η} then η cannot compute it directly. Instead we introduce four different approximations to the DU that vary in their balance between learnability and factoredness. These four utilities are named so that the first two letters of the utility represent how the two terms of the difference utility deal with partial observability. "B" stands for "broadcast" meaning that the world utility is broadcast to the system, "T" stands for "truncated" meaning that the hidden values are ignored, and "E" stands for "estimated" meaning that the hidden variable is estimated from the observed variables. Table I shows the factoredness, learnability and communication level trade-offs for DU and each of the four utilities presented below (e.g., BEU is fully factored, has low learnability and uses local communications as well as global broadcasts, whereas EEU is partially factored, has high learnability and only uses local communications).

1) Broadcast/Truncated Utility (BTU): BTU is a variant of DU, where the communication restrictions force agent η to set not only its own state, but also the states of all agents that it cannot observe to the null state:

$$BTU_\eta(z) \quad = \quad G(z) - G(z - z^{h_\eta} - z_\eta) \tag{4}$$

Note that BTU, as well as BEU (discussed below), assume that the true world utility can be broadcast despite the communication restriction. In many applications, this is a assumption is reasonable since the world utility can often be computed once and broadcast throughout the environment [9]. More complex forms of broadcasting are often used for distributed multi-agent systems [5], but in this paper we will assume a very simple global broadcast of a single number.

Despite creating a virtual state by setting more than η to the null state, BTU is still factored since it is in the form of the difference utility (e.g., the second term of Equation 4 does not depend on η). However, this utility generally has significantly more noise than a pure DU since the difference removes not only η's contribution, but all states hidden from η. Accordingly, in situations where a large number of agents are hidden from η, BTU suffers from poor signal to noise problems, e.g., at the limit of agent η observing only its own actions, the second term becomes $G(\vec{0})$.

2) Truncated/Truncated Utility (TTU): The second private utility is conceptually similar to BTU except that both terms are computed under the communication restrictions:

$$TTU_\eta(z) \quad = \quad G(z - z^{h_\eta}) - G(z - z^{h_\eta} - z_\eta). \tag{5}$$

Essentially, TTU is DU where z is approximated by $z - z^{h_\eta}$. Because of this, TTU is not factored with respect to the world utility $G(z)$. While not being factored with world utility, TTU generally has higher learnability than BTU [20].

Again, consider the case where a large number of agents, not interacting with η, are hidden from η. The contribution of those agents will not be included in either term of TTU, since both terms are computed with the communication restriction. Therefore this utility will have less noise. However, if the assumption that $G(z - z^{h_\eta})$ is close to $G(z)$ does not hold (e.g., some hidden agents are crucial to the system's behavior) then TTU will not produce good system performance.

3) Broadcast/Estimated Utility (BEU): The third utility is similar to BTU, except that instead of truncating the components of z^{h_η} (e.g., setting them to zero), their values are estimated given the values of z^{o_η}:

$$BEU_\eta(z) \quad = \quad G(z) - G(z^{o_\eta} + E[z^{h_\eta}|z^{o_\eta}] - z_\eta) \tag{6}$$

where $E[z^{h_\eta}|z^{o_\eta}]$ gives the expected hidden state given the states observable to η. As long as this estimate is not influenced by the actions of η beyond z_η, this utility is factored, since the first term of the difference equation is still $G(\eta)$. While both BTU and BEU are factored, BEU may have less noise, depending on how good the estimate for z^{h_η} is.

Again, consider a system where a large number of agents that do not interact with η that are hidden from η's state, but that their values can be approximated from the visible components of the state. In this case the first term of BEU will contain the agents' contribution to $G(z)$, but the second term will subtract out their inferred contribution. Even if effects of the hidden elements cannot be perfectly estimated, significant amounts of noise can be eliminated from the system. Note however that if the estimate is particularly poor, noise can also be introduced into the system.

4) Estimated/Estimated Utility (EEU): The fourth utility is similar to TTU, except that instead of truncating the hidden elements, the value of z^{h_η} is estimated in both terms:

$$\begin{aligned} EEU_\eta(z) \quad = \quad & G(z^{o_\eta} + E[z^{h_\eta}|z^{o_\eta}]) - \\ & G(z^{o_\eta} + E[z^{h_\eta}|z^{o_\eta}] - z_\eta). \end{aligned} \tag{7}$$

Essentially, EEU is a DU where z is approximated by $z^{0_\eta} - E[z^{h_\eta}|z^{o_\eta}]$. As was the case with TTU this utility is not factored with respect to the world utility G. However, with a good estimate of z^{h_η}, the value $G(z^{o_\eta} - E[z^{h_\eta}|z^{o_\eta}])$ will be much closer to $G(z)$ than $G(z^{o_\eta})$, so this utility can be much closer to being factored with respect to $G(z)$ than can TTU.

In addition this utility retains TTU's desirable property that both terms are using the same version of the state. Since both terms are estimating the values of z^{h_η} in the same way, any contribution that the non-η terms of z^{h_η} make on the first term will be subtracted out in the second term. Note that unlike with BEU, even if the estimate of the hidden components is very poor, noise will not be added to the system since both terms of the utility use the same estimate. Instead, the quality of the estimate only affects how close this utility is to being factored with respect to $G(z)$.

III. Congestion Games

Congestion games are characterized by having the world utility depend on the agents use of a particular resource (e.g., quality of an agent's action depends on the number of other agents selecting the same action) [2], [11]. This type of problem arises in many domains, ranging from telecommunications (e.g., response of a link depends on the number of users), transportation (e.g., value of a highway lane depends on the number of cars), power/computer grids (e.g., performance of a server depends on the number of scheduled jobs), and public good distribution (e.g., enjoyment of a park/restaurant depends on the number of people using it). In each instance of the problem, at each time step, each agent η has to decide whether to participate (e.g., use server, drive on a lane, attend restaurant) in the use of that resource or not. The nature of the problem produces a "congestion" (e.g., if most agents believe the resource will be under-used, they will use it and cause it to be over-used, and vice-versa).

In this work, we focus on the following instantiation of the congestion game: There are N agents, each picking one out of K actions each time step. Those actions result in a world utility, G, given by:

$$G(z) \equiv \sum_{k=1}^{K} x_k(z)\, e^{\frac{-x_k(z)}{c_k}} \;, \tag{8}$$

where $x_k(z)$ is the number of agents choosing action k; z_η is η's choice at that time step; and c_k is the optimal "capacity" of resource k. At the end of the time step, the associated private utilities for each agent are communicated to that agent, and the process is repeated.

Since we wish to concentrate on the effects of the utilities rather than on the algorithms that use them, we use a very simple learning algorithm, though a number of learning methods (e.g., neural networks, Q-learning) can be used. In this simple algorithm each agent η keeps a K-dimensional vector giving its estimates of the utility it would receive for choosing that action. The decisions are made using the vector, with an ϵ-greedy learner with ϵ set to 0.05. All of the vectors are initially set to zero and there is a learning rate decay of 0.99.

A. Communication Restrictions

We model communication restrictions in this problem by controlling how many other agents one agent can "talk" to. Without this communication the agent cannot know what the other agents have done. We define a communication level B in the range $[0.0, 1.0]$ representing the fraction of all the agents to which an agent can talk. When $B = 1.0$ an agent can talk to the all other agents, whereas when $B = 0.0$ an agent has no communication, and thus is only aware of its own action. In this problem, communication restrictions result in variations on how $x_k(z)$ is computed. For truncated versions of the DU, (BTU and TTU), η uses $x_k(z^{o_\eta})$, which provides the number of observable agents that have selected action k. (note since in BTU the first term is broadcast, the agent does not need to compute it). For utilities using an estimate of the state (BEU and EEU), $x_k(z^{o_\eta})$ is scaled, and $\frac{1}{B}x_k(z^{o_\eta})$ represents agent η's estimate of how many agents selected action k. Note this is an extremely simple estimation procedure and does not take any information an agent collects to modify how it forms this estimate.

IV. Experimental Results

We tested the performance of the four versions of the DU with varying levels of communication. The tests were conducted in a congestion game with 100 agents and with $c_k = 5$ for all k. All of the trials were conducted for 1000 episodes, and were run 25 times.

Figure 2 shows the performance of the four utilities with different levels of communication. When the communication level is high, the utilities converge to DU. When communication is very low, the BTU and BEU have the best performance because their first term, G, is not affected by the communication restriction. They essentially are reduced to a team game, and give moderately good performance. Note that the performance of BTU is worse at 50% communication than at 5%. This counterintuitive result is explained by how the utility is computed in this problem. With little communication, the total number of agents that can be seen is small, and the contribution of the second term is small. In contrast, with 50% communication, the second term will be large enough to have an impact on the utility. However, because both at 5% and 50% communication levels $x_k(z^{o_\eta})$ is significantly different than $x_k(z)$, neither provides a usable second term. In fact, rather than subtracting out noise, the second term adds noise.

For most levels of communication restriction, the EEU performs the best and performs up to 75% closer to optimal than

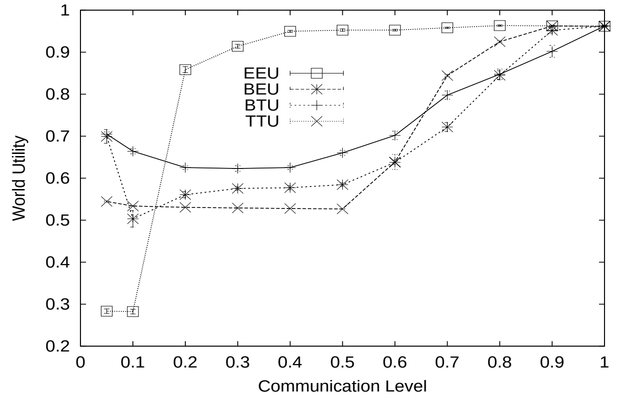

Fig. 2. Performance of four utility functions for a range of communication levels. For moderate communication levels EEU performs best. For very low communication BTU performs best since, it uses information from world utility.

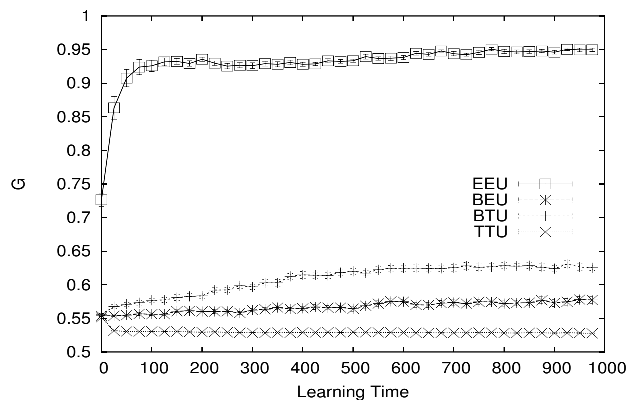

Fig. 3. Learning rates of four utility functions at 40% communication. EEU learns far more quickly than the others, because it provides a cleaner signal. Note that even though TTU is highly learnable, it is not close to being factored with respect to G, so it has a flat learning curve. Both BEU and BTU learn because they are factored, but because they have low learnability (too much noise in the signal) their learning curve is extremely slow.

utilities which use the same information. Recall that EEU and TTU are not factored, whereas BTU and BEU are. What helps EEU in this case is that although it is not factored, as long as the estimate for G in the first term is sufficiently close to G, it is close to being factored. Furthermore, because both the first and second terms use the same estimate for the state, the subtraction does remove noise, as intended. The utility TTU performs the worst even though there may not be much noise in the utility. This low performance is caused by TTU being far from factored due to the truncation of the hidden state components.

Figures 3 and 4 give a clearer view of the performances at a fixed level of communication restriction (40% and 70% respectively). EEU is clearly superior at 40% communication. it is close to being factored and because of its high learnability it rapidly converges to a good solution. Both BTU and BEU are factored, but suffer from significantly low learnability. At 70% communication TTU displays the problem with utilities that are not factored: the more the agents learn the worse

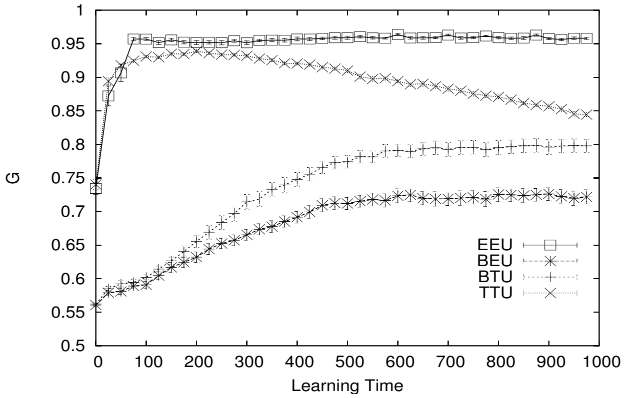

Fig. 4. Learning rates of four utility functions at 70% communication. The difference between the factored EEU and non-factored TTU is more explicit in this case. TTU does well initially, but as the agent continue to learn its performance suffers. This is because this system is nonfactored. This means the agents doing well on their own utilities can (and in this case, do) hurt system performance. Furthermore, because TTU has high learnability, agents succesfully learn to do the wrong thing. Both BEU and BTU performance improves with learning because both are factored, but because they have low learnability (too much noise in the signal) their learning curve is extremely slow and flattens out before reaching a good performance level.

the system performance becomes. Because this system is not factored (or in this case, not close to being factored) the agents optimizing their private utilities do not optimize the world utility. Ironically, because TTU has good learnability (i.e., the slope of TTU shows no sign of flattening out at $t = 1000$) the agents learn to do the wrong thing successfully. BTU and BEU on the other hand are factored so G does not decrease. However, because of learnability issues, after an initial period of improvement, the agents encounter a difficult signal to noise problem and the system performance stops improving.

V. Discussion

In this work we focus on the problem of designing a collective of autonomous agents in the presence of significant communication restrictions. In such cases, private utilities that rely on agents having access to a fully connected communication network may break down. We presented four different utility functions that each make different tradeoffs among what communication is available to an agent and how that information is used. We showed that in an instance of a congestion game, one of the utilities, EEU, does significantly better than all the others. Agents using this utility learn faster and achieve better results in our experiments. Furthermore this analysis shows a tradeoff between using world utility broadcast or not. For very low levels of communication (e.g., under 10%) using the global broadcast is beneficial (e.g, BTU). For all other cases, balancing the way in which the utility is computed by using the same state estimates in both terms of the DU provides the best solutions (EEU).

We are also currently exploring the benefits of team formation in helping overcome communication restrictions in a collective. Future work in this area includes investigating new utility functions for the agents, dynamic team formation where agents may join and/or leave teams in an adaptive fashion, and incurring a cost for sharing information. Furthermore we are determining the effectiveness of using the utilities as fitness evaluation functions for evolutionary computation with neural networks [1].

References

[1] A. Agogino and K. Tumer. Efficient evaluation functions for multi-rover systems. In *The Genetic and Evolutionary Computation Conference*, Seatle, WA, June 2004.

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

[3] Tucker Balch and Ronald C. Arkin. Communication in reactive multi-agent robotic systems. *Autonomous Robots*, 1(1):27–52, 1994.

[4] L. Blumrosen and N. Nisan. Auctions with severely bounded communication. In *The 43rd Annual IEEE Symposium on Foundations of Computer Science*, Vancouver, Canada, November 2002.

[5] P. Busetta, A. Dona, and M. Nori. Channeled multicast for group communications. In *Proceedings of the First International Joint Conference on Autonomous Agents and Multi-Agent Systems*, pages 1280–1287, Bologna, Italy, July 2002.

[6] 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.

[7] M. Dastani, J. van der Ham, and F. Dignum. Communication for goal directed agents. In *Proceedings of the Agent Communication Languages and Conversation Policies*, Bologna, Italy, July 2002.

[8] J. Fredslund and M. J. Mataric. Robots in formation using local information. In *Proceedings, 7th International Conference on Intelligent Autonomous Systems (IAS-7)*, pages 100–107, Marina del Rey, CA, March 2002.

[9] D. Gage. How to communicate with zillions of robots. In *Proceedings of SPIE Mobile Robots V III*, pages 250–257, Boston, MA, 1993.

[10] T. Groves. Incentives in teams. *Econometrica*, 41:617–631, 1973.

[11] P. Jefferies, M. L. Hart, and N. F. Johnson. Deterministic dynamics in the minority game. *Physical Review E*, 65 (016105), 2002.

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

[13] D. Pynadath and M. Tambe. The communicative multiagent team decision problem: Analyzing teamwork theories and models. *Journal of Artificial Intelligence Research*, 16:389–423, 2002.

[14] Ira A. Smith and Philip R. Cohen. Toward a semantics for a speech act based agent communications language. In Tim Finin and James Mayfield, editors, *Proceedings of the CIKM '95 Workshop on Intelligent Information Agents*, Baltimore, Maryland, 1995.

[15] 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*, pages 378–385, Bologna, Italy, July 2002.

[16] K. Tumer and J. Lawson. Collectives for multiple resource job scheduling across heterogeneous servers. In *Proceedings of the Second International Joint Conference on Autonomous Agents and Multi-Agent Systems*, July 2003.

[17] K. Tumer and D. Wolpert. A survey of collectives. In *Collectives and the Design of Complex Systems*. Springer, 2004. To appear.

[18] 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.

[19] W. Vickrey. Counterspeculation, auctions and competitive sealed tenders. *Journal of Finance*, 16:8–37, 1961.

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

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

[22] P. Xuan, V. Lesser, and S. Zilberstein. Communication decisions in multi-agent cooperation: Model and experiments. In *Proceedings of the Fifth International Conference on Autonomous Agents*, pages 616–623, Montreal, January 2001. ACM Press.