A SURVEY OF COLLECTIVE INTELLIGENCE

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

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

March 4, 1999

Abstract

Spell-check. check "personal" vs. "local" used consistently. Check the discussion in reinf.learning.c in regard to the definition used here for utility functions. Get reference to Ann's paper into the El Farol section.

This chapter presents the science of "COllective INtelligence" (COIN). A COIN is a large multi-agent systems where:
i) the agents each run reinforcement learning (RL) algorithms;
ii) there is little to no centralized communication or control;
iii) there is a provided world utility function that rates the possible histories of the full system.

The conventional approach to designing large distributed systems to optimize a world utility does not use agents running RL algorithms. Rather that approach begins with explicit modeling of the overall system's dynamics, followed by detailed hand-tuning of the interactions between the components to ensure that they "cooperate" as far as the world utility is concerned. This approach is labor-intensive, often results in highly non-robust systems, and usually results in design techniques that have limited applicability.

In contrast, with COINs we wish to solve the system design problems implicitly, via the 'adaptive' character of the RL algorithms of each of the agents. This COIN approach introduces an entirely new, profound design problem: Assuming the RL algorithms are able to achieve high rewards, what reward functions for the individual agents will, when pursued by those agents, result in high world utility? In other words, what reward functions will best ensure that we do not have phenomena like the tragedy of the commons, or Braess's paradox?

Although still very young, the science of COINs has already resulted in successes in artificial domains, in particular in packet-routing, the leader-follower problem, and in variants of Arthur's "El Farol bar problem". It is expected that as it matures not only will COIN science expand greatly the range of tasks addressable by human engineers, but it will also provide much insight into already established scientific fields, such as economics, game theory, or population biology.

1 INTRODUCTION

Over the past decade or so two developments have occurred in computer science whose intersection promises to open a vast new area of research, an area extending far beyond the boundaries of conventional computer science. The first of these developments is the growing realization of how useful it would be to be able to control distributed systems which have little (if any) centralized communication, and to do so 'adaptively', with minimal reliance on detailed knowledge of the system's small-scale dynamical behavior. This realization has been most recently manifested in the field of amorphous computing []. The second development is the maturing of the discipline of reinforcement learning (RL). This is the branch of machine learning that is concerned with an agent who periodically receives 'reward' signals from the environment that partially reflect the value of that agent's personal utility function. The goal of RL is to determine how, using those reward signals, the agent should update its action policy so as to maximize its utility [115, 203, 214].

Intuitively, one might hope that the tool of RL would help us solve the distributed control problem, since RL is adaptive, and in particular since it is not restricted to domains having sufficient breadths of communication. However by itself, conventional single-agent RL does not provide a means for controlling large, distributed systems. This is true even if the system *does* have centralized communication. The problem is that the space of possible action policies for such systems is too big to be searched. We might imagine as a variant using a large set of agents, each controlling only part of the system. Since the individual action spaces of such agents would be relatively small, we could realistically deploy conventional RL on each one. However now we face the central question of how to map the world utility function concerning the overall system into personal utility functions for each of the agents. In particular, how should we design those personal utility functions so that each agent can realistically hope to optimize its function, and at the same time the collective behavior of the agents will optimize the world utility?

We use the term "COllective INtelligence" (COIN) to refer to any pair of a large, distributed collection of interacting RL algorithms among which there is little to no centralized communication or control, together with a world utility function that rates the possible dynamic histories of the collection. The central COIN design problem is how, without any detailed modeling of the overall system, you can set the utility functions for the RL algorithms in a COIN to have the overall dynamics reliably and robustly achieve large values of the provided world utility. The benefits of an answer to this question would extend beyond the many branches of computer science, having major ramifications for many other sciences as well. The next section discusses some of those benefits. The following section reviews previous work that has bearing on the COIN design problem. The final section constitutes the core of this chapter. It presents a quick outline of a promising mathematical framework for addressing this problem, and then experimental illustrations of the prescriptions of that framework. Throughout we will use italics for emphasis, single quotes for informally defined terms, and double quotes to delineate colloquial terminology.

2 Background

There are many design problems that involve distributed computational systems where there are strong restrictions on centralized communication ('we can't all talk'); or there is communication with a central processor, but that processor is not sufficiently powerful to determine how to control the entire system ('we aren't smart enough'); or the processor is powerful enough in principle, but it is not clear what algorithm it could run by itself that would effectively control the entire system ('we don't know what to think'). Just a few of the potential examples include:

i) Designing a control system for constellations of communication satellites or of constellations of planetary exploration vehicles (world utility in the latter case being some measure of quality of scientific data collected);

ii) Designing a control system for routing over a communication network (world utility being some aggregate quality of service measure);

iii) Construction of parallel algorithms for solving numerical optimization problems (the optimization problem itself constituting the world utility);

iv) Vehicular traffic control, e.g., air traffic control, or high-occupancy-toll-lanes for automobiles. (In these problems the individual agents are humans and the associated utility functions must be of a constrained form, reflecting the relatively inflexible kinds of preferences humans possess.);

v) Routing over a power grid;

vi) Control of a large, distributed chemical plant;

viii) Control of the elements of an amorphous computer;

ix) Control of the elements of a 'noisy' phased array radar;

x) Compute-serving over an information power grid.

Such systems may be best controlled with an artificial COIN. The potential usefulness of solving the COIN design problem extends far beyond such engineering concerns however. That's because the COIN design problem is an inverse problem, whereas essentially all of the scientific fields that are concerned with naturally-occurring distributed systems analyze them purely as a "forward problem". That is, those fields analyze what global behavior would arise from provided local dynamical laws, rather than grapple with the inverse problem of how to configure those laws to induce desired global behavior. It seems highly likely that the insights garnered from understanding the inverse problem would provide a trenchant novel perspective on those fields. Just as tackling the inverse problem in the design of steam engines lead to the first true understanding of the macroscopic properties of physical bodes (aka thermodynamics), so may the cracking of the COIN design problem hopefully would augment our understanding of many naturally-occurring COINs.

As an example, consider countries with capitalist human economies. Such systems can be viewed as naturally occurring COINs. One can declare 'world utility' to be a time average of the Gross Domestic Product (GDP) of the country in question. (World utility per se is not a construction internal to a human economy, but rather something defined from the outside.) The reward functions for the human agents are the achievements of their personal goals (usually involving personal wealth to some degree). As commonly understood, the economy of the United Stated in the 1990's, or of Japan through much of the 1970's and 1980's, serves as an existence proof that the COIN design problem has solutions.

Now in general, to achieve high global utility in a COIN it is necessary to avoid having the agents work at cross-purposes, lest phenomena like the Tragedy of the Commons (TOC) occur, in which individual avarice works to lower global utility [91]. One way to avoid such phenomena is by modifying the agents' utility functions. In the context of capitalist economies, this can be done via punitive legislation. A real world example of an attempt to make just such a modification was the creation of anti-trust regulations designed to prevent monopolistic practices.

In designing a COIN we usually have more freedom than anti-trust regulators though, in that there is no base-line "organic" local utility function over which we must superimpose legislation-like incentives. Rather, the entire "psychology" of the individual agents is at our disposal, when designing a COIN. This obviates the need for honesty-elicitation ('incentive compatible') mechanisms, like auctions, which form a central component of conventional economics. Accordingly, COINs can differ in certain crucial respects from human economies. The precise differences — the subject of current research — seem likely to present many insights into the functioning of economic structures like anti-trust regulators.

Another example of the novel perspective of COINs, also concerning human economies, is the usefulness of (commodity, or especially fiat) money. The traditional economics view is that money is useful because it is portable; universally valued (and therefore minimizes the number of "trading posts" needed [200]); allows "middlemen" to facilitate resource allocation, and the like. The COIN perspective however leads us to address lower-level aspects of the usefulness of money. For example, formally, 'money' constitutes a particular class of couplings between the states and utility functions of the various agents. Now for any underlying sysstem any particular choice of utility functions for the agents —likeutility functions involving money — will induce high levels of some world utilities. But it will simultaneously induce *low levels* of world utilities. This raises a host of questions, like how to formally specify the most general set of world utilities which benefit significantly from money-based local utility functions form the class of such functions involving money. If one is provided a world utility that is not a member of that set, then an "economics-inspired" configuration of the system is likely to result in poor performance.

There are many other scientific fields which are currently under investigation from a COIN-design perspective. Some of them are, like economics, part of (or at least closely related to) the social sciences. These fields typically involve RL algorithms under the guise of human agents. (An example is game theory, especially game theory of bounded rational players.)

Of course, real-world economies are "emergent" and don't have externally imposed world utilities, like time-average of GDP. Rather such utilities are an analytic tool that an understanding of COINs would exploit to gain insight into the functioning of human economies. There are other scientific fields that might benefit from a COIN-design perspective even though they study systems that don't even involve RL algorithms. The idea here is that if we viewed such systems from a teleological perspective, both in concentrating on a world utility and in casting the nodal elements of the system as RL algorithms, we could learn a lot about the form of the 'design space' in which such systems live. Examples here are ecosystems (individual genes, individuals, or species being the nodal elements) and cells (individual organelles in Eukaryotes being the nodal elements). In both cases, the world utility could involve robustness of the desired equilibrium against external perturbation, efficient exploitation of free energy in the environment, etc.

3 Review of Literature Related to COINs

There are many different features that characterize what we mean by a "COIN". The first four features in the following list are definitional; the remainder are not definitional per se, but are fundamental to the sorts of COINs we are concerned with in this chapter.

1) There are many processors running concurrently, performing actions that affect one another's behavior.

2) There is little to no centralized personalized communication, i.e., little to no behavior in which a small subset of the processors communicates with all the other processors, but communicates differently with each one of those other processors. Any single processor's "broadcasting" the same information to all other processors is not precluded.

3) There is little to no centralized personalized control, i.e., little to no behavior in which a small subset of the processors controls all the other processors, but controls each one of those other processors differently. "Broadcasting" the same control signal to all other processors is not precluded.

4) There is a well-specified task, typically in the form of extremizing a utility function, that concerns the behavior of the entire distributed system. So we are confronted with the inverse problem of how to configure the system to achieve the task.

5) The individual processors are running RL algorithms.

6) The approach for tackling (4) is scalable to very large numbers of processors.

7) The approach for tackling (4) is very broadly applicable. In particular, it can work when little (if any) "broadcasting" as in (2) and (3) is possible.

8) The approach for tackling (4) involves little to no hand-tailoring.

9) The approach for tackling (4) is robust and adaptive, with minimal need to "get the details exactly right or else", as far as the stochastic dynamics of the system is concerned.

The rest of this section reviews some of the fields that are related to COINs, and in particular characterizes them in terms of this list of nine characteristics of COINs.

3.1 AI and Machine Learning

3.1.1 Reinforcement Learning

As discussed in the introduction, the maturing field of reinforcement learning provides a much needed tool for the types of problems addressed by COINs. Because RL generally provides model-free[1] and "online" learning features, it is ideally suited for the distributed environment where a "teacher" is not available and the agents need to learn successful strategies based on "rewards" and "penalties" they receive from the overall system at various intervals. It is even possible for the learners to use those rewards to modify *how* they learn [186].

Although work on RL dates back to Samuel's checker player [180], relatively recent theoretical [214] and empirical results [207, 52] have made RL one of the 'hottest' areas in machine learning. Many problems ranging from controlling a robot's gait to controlling a chemical plant to allocating constrained resource have been addressed with considerable success using RL [89, 107, 159, 173, 228]. In particular the RL algorithms $TD(\lambda)$ (which rates potential states based on a *value function*) [203] and Q–learning (which rates action-state pairs) [214] have been investigated extensively. A detailed investigation of RL is available in [115, 204, 214].

Although powerful and widely applicable, solitary RL algorithms will not perform well on large distributed heterogenous problems in general. This is due to the very big size of the action-policy space for such problems. In addition, without centralized communication and control, how a solitary RL algorithm could run the full system at all, poorly or well, becomes a major concern.[2] For these reasons, it is natural to consider deploying many RL algorithms rather than a single one for these large distributed prob-

[1] There exist some model-based variants of traditional RL. See for example [6].

[2] One possible solution would be to run the RL off-line on a simulation of the full system and then convey the results to the components of the system at the price of a single centralized initialization (e.g., [158]). In general though, this approach will suffer from being extremely dependent on "getting the details right" in the simulation.

lems. We will discuss the coordination issues such an approach raises in conjuction with multi-agent systems in Section 3.1.3 and with learnability in COINs in Section 4.

3.1.2 Distributed Artificial Intelligence

The field of Distributed Artificial Intelligence (DAI) has arisen as more and more traditional Artificial Intelligence (AI) tasks have migrated toward parallel implementation. The most direct approach to such implementations is to directly parallelize AI production systems or the underlying programming languages [75, 178]. An alternative and more challenging approach is to use distributed computing, where not only are the individual reasoning, planning and scheduling AI tasks parallelized, but there are *different modules* with different such tasks, concurrently working toward a common goal [111, 134, 112].

In a DAI, one needs to ensure that the task has been modularized in a way that improves efficiency. Unfortunately, this usually requires a central controller whose purpose is to allocate tasks and process the associated results. Moreover, designing that controller in a traditional AI fashion often results in brittle solutions. Accordingly, recently there has been a move toward both more autonomous modules and fewer restrictions on the interactions among the modules. **Kagan: References here?**

Despite this evolution, DAI maintains the traditional AI concern with a pre-fixed set of *particular* aspects of intelligent behavior (e.g. reasoning, understanding, learning etc.) rather than on their *cumulative* character. As the idea that intelligence may have more to do with the interaction among components started to take shape, focus shifted to new concepts that better incorporated that idea [37, 38, 113]. However such systems tend to be hand-tailored, and have associated scalability problems. **Kagan: are we sure these are their major deficiencies?**

3.1.3 Multi-Agent Systems

The field of Multi-Agent Systems (MAS) is concerned with the interactions among the members of such a set of agents [113, 190, 205, 85, 35], as well as the inner workings of each agent in such a set (e.g., their learning algorithms) [31, 32, 33]. As in computational ecologies and computational markets (see below), a well-designed MAS is one that achieves a global task through the actions of its components. The associated design steps involve [113]:

1. Decomposing a global task into distributable subcomponents, yielding tractable tasks for each agent;
2. Establishing communication channels that provide sufficient information to each of the agents for it to achieve its task, but are not too unwieldly for the overall system to sustain; and

3. Coordinating the agents in a way that ensures that they cooperate on the global task, or at the very least does not allow them to pursue conflicting strategies in trying to achieve their tasks.

Step (3) is rarely trivial; one of the main difficulties encountered in MAS design is that agents act selfishly and artificial cooperation structures have to be imposed on their behavior to enforce cooperation [9]. An active area of research is to determine how selfish agents' "incentives" have to be engineered in order to avoid the TOC [194]. When simply providing the right incentives is not sufficient, one can resort to strategies that actively induce agents to cooperate rather than act selfishly. In such cases negotiations [125], coalition formation [183, 182] or contracting [2] among agents may be needed to ensure that they do not work at cross purposes.

Unfortunately, all of these approaches share with DAI and its offshoots the problem of relying excessively on hand-tailoring, and therefore being difficult to scale and often non-robust. In addition, except as noted in the next subsection, they involve no RL.

3.1.4 Reinforcement Learning-Based Multi-Agent Systems

Because it neither requires explicit modeling of the environment nor having a "teacher" that provides the "correct" actions, the approach of having the individual agents in a MAS use RL is well-suited for MAS's deployed in domains where one has little knowledge about the environment and/or other agents. There are two main approaches to designing such MAS's:
(i) One has 'solipsistic agents' which don't know about each other and whose RL rewards are given by the performance of the entire system (so the joint actions of all other agents form an "inanimate background" contributing to the reward signal each agent receives);
(ii) One has 'social agents' that explicitly model each other and take each others' actions into account.

Both (i) and (ii) can be viewed as ways to (try to) coordinate the agents in a MAS in a robust fashion.

Solipsistic Agents: MAS's with solipsistic agents have been successfully applied to a multitude of problems [88, 97, 181, 185, 52]. Generally these schemes use RL algorithms similar to those discussed in Section 3.1.1. However much of this work lacks a well defined global task or broad applicability (e.g., [181]). More generally, none of the work with solipsistic agnets scales well. The problem is that each agent must be able to discern the effect of its actions on the overall performance of the system, since that performance constitutes its reward signal. As the number of agents increases though, the effects of any one agent's actions (signal) will be swamped by the effects of other agents (noise), making the agent unable to learn well, if at all. (See the discussion below on learnability.) In addition, of course, solipsistic agents cannot be used in situations lacking centralized calculation and broadcast of the single global reward signal.

Social agents: MAS's whose agents take the actions of other agents into account synthesize RL with game theoretic concepts (e.g., Nash equilibrium). They do this to try to ensure that the overall system both moves toward achieving the overall global goal and avoids oscillatory behavior [51, 77, 106, 105]. To that end, the agents incorporate internal mechanisms that actively model the behavior of other agents. In Section 3.2.5 we discuss a situation where such modeling is necessarly self-defeating. More generally, this approach suffers from being narrowly applicable, requiring hand tailoring, and potentially not scaling well.

Move [116, 225, 73] to game theory section?

3.2 Social Science - Inspired Systems

Economics provides more than examples of naturally occurring systems that can be viewed as a (more or less) well-performing COINs. Both empirical economics (e.g., economic history, experimental economics, auction methods **Kagan: ref.'s?, auction methods are empirical in what sense?**) and theoretical economics (e.g., general equilibrium theory, theory of optimal taxation **Kagan: ref.'s?**), provide a rich literature on how to study strategic situations where many parties interact, rationally or otherwise. **Kagan: Really? Theoretical work on subrational situations?**

In this section we summarize the two economics concepts that are probably the most closely related to COINs, in that they deal with how a large number of interacting agents can function in a stable and efficient manner: general equilibrium theory and mechanism design. We then discuss general attempts to apply those concepts to distributed computational problems. We follow this with a discussion of game theory, and then present a particular celebrated toy-world problem that involves many of these issues.

3.2.1 General Equilibrium Theory

Often the first version of "equilibrium" that one encounters in economics is that of supply and demand in single markets: the price of the market's good is determined by where the supply and demand curves for that good intersect. In cases where there is interaction among multiple markets however, one cannot simply determine the price of each market's good individually, as both the supply and demand for each good depends on the supply/demand of other goods. Considering the price fluctuations across markets leads to the concept of 'general equilibrium', where prices for each good are determined in such a way to ensure that all markets 'clear'. Intuitively, this means that prices are set so the total supply of each good is equal to the demand for that good [3, 199]. [3] The existence of such an equilibrium was first postulated by Leon Walras [213]. A

[3] More formally, each agent's utility is a function of that agent's allotment of all the possible goods. In addition, every good has a price. (Utility functions are independent of money.) Therefore every player has an 'budget', given by their intial allotment of goods and the associated initial prices. We pool all

mechanism that calculates the equilibrium (i.e., market-clearing) prices now bears his name: the Walrasian auctioner.

One of the major shortcomings of general equilibrium theory is that it does not readily accommodate the concept of money [76]. Of the three main roles money plays in an economy (medium of exchange in trades, store of value for future trades, and unit of account) none are essential in a general equilibrium setting. The unit of account aspect is not needed as the bookkeeping is performed by the Walrasian auctioner. Since the supplies and demands are matched directly there is no need to facilitate trades, and thus no role for money as a medium of exchange. And finally, as the system reaches an equilibrium in one step, through the auctioner, there is no need to store value for future trading rounds [138].

The reason that money is not needed can be traced to the fact that there is an "overseer" with global information who guides the system. If we remove the centralized communication and control exerted by this overseer, then (as in a real economy) agents will no longer know the exact details of the overall economy. They will be forced to makes guesses as in any learning system, and the resultant differences in their actions may thereby be amplified [128, 129]. **Kagan: "differences" between what and what?**

Such a decentralized learning-based system more closely resembles a COIN than does a general equilibrium system. In contrast to general equilibrium systems, the three main roles money plays in a human economy are crucial to the dynamics of such a decentralized system [10]. This comports with the important effects in COINs of having the agents' utility functions involve money (see Background section above).

3.2.2 Mechanism Design

The field of mechanism design encompasses auctions, monopoly pricing, optimal taxation and public good theory [126]. It is concerned with the incentives that must be applied to any set of agents that interact and exchange goods[209, 153] in order to get those agents to exhibit desired behavior. Usually that desired behavior concerns pre-specified utility functions of some sort for each of the individual agents. In particular, mechanism desgin is concerned with 'efficient' incentive schemes which ensure that all bidders in an auction "benefit" from the outcome, and 'optimal' incentive schemes which maximize the revenue of the involved parties.

One particularly important type of such incentive schemes is auctions. When many agents interact in a common environment often there needs to be a structure that supports the exchange of goods or information among those agents. Auctions provide one

the agents' goods together. Then we set prices for all of those goods, and allocate the goods back among the agents in such a way that each agent is given a total value of goods (as determined by the prices) equal to that agent's budget. 'Markets clear' at those prices for which each agent views its allocation of goods as optimizing its utility, subject to its budget and to those prices for the goods.

such (centralized) structure for managing exchanges of goods. For example, in the English auction all the agents come together and 'bid' for a good, and the price of the good is increased until only one bidder remains, who gets the good in exchange for the resource bid. As another example, in the Dutch auction the price of a good is decreased until one buyer is willing to pay the current price.

All auctions perform the same task: match supply and demand. As such, auctions are one of the ways in which price equilibriation among a set of interacting agents (perhaps an equilibriation approximating general equilibrium, perhaps not) can be achieved. However, the mechanisms used in auction are not necessarily efficient. For example, the winner of an English auction may well have been willing to pay more for the good. **Kagan: How is this inefficient, as opposed to non-optimal?**

3.2.3 Computational Economics

'Computational economies' are economics-inspired schemes for managing the components of a distributed computational system, which work by having a 'computational market' guide the interactions among those components. Such a market is defined as any structure that allows the components of the system to exchange information on relative valuation of resources (as in an auction), establish equilibrium states (e.g., determine market clearing prices) and exchange resources (i.e., engage in trades).

Such computational economies can be used to investigate real economies and biological systems [120, 30, 29, 26]. They can also be used design distributed computational systems. For example, such computational economies are well-suited to many distributed resource allocation problems, where each component of the system can either directly produce the "goods" it needs or acquire them through trades with other components. Computational markets often allow for far more heterogeneity in the components than do conventional resource allocation schemes. Furthermore, there is both theoretical and empirical evidence suggesting that such markets are often able to settle to equilibrium states. For example, auctions find prices that satisfy both the seller and the buyer which results in an increase in the utility of both (else one or the other would not have agreed to the sale). Assuming that all parties are free to pursue trading opportunities, such mechanisms move the system to a point where all possible bilateral trades that could improve the utility of both parties are exhausted. Such a state of the system where any change that increases the utility of one agent must decrease the utility of another is called 'Pareto optimal' citesomething. It constitutes a local maximum of the global utility function [208]. **Kagan: huh? This needs elaborating.**

discuss why this is desirable

$\Longleftarrow$

One example of such a computational economy being used for resource allocation is Huberman and Clearwater's use of a double–blind auction to solve the complex task of controlling the temperature of a building. In this case, each agent (individual temperature controller) bids to buy or sell cool or warm air. This market mechanism leads to an equitable temperature distribution in the system [109]. Other domains where market mechanisms were successfully applied include purchasing memory in an operating systems [47], allocating virtual circuits [72], "stealing" unused CPU cycles in a network of computers [65, 211], predicting option futures in financial markets [172], and numerous scheduling and distributed resource allocation problems [127, 133, 195, 201, 217, 218].

Computational economics can also be used for tasks not tightly coupled to resource allocation. For example, following the work of Maes [142] and Ferber [71], Baum shows how by using computational markets a large number of agents can interact and cooperate to solve a variant of the blocks world problem [18, 19, 20]

Viewed as candidate COINs, all market-based computational economics fall short in relying on both centralized communication and centralized control to some degree. Often that reliance is extreme. For example, the the systems investigated by Baum not only have the centralized control of a market, but in addition have centralized control of all other non-market aspects of the system. (Indeed, the market is secondary, in that it is only used to decide which single expert among a set of candidate experts gets to exert that centralized control at any given moment). There has also been doubt cast on how well computational economies perform in practice [208], and they also often require extensive hand-tailoring in practice.

$\Longleftarrow$

3.2.4 Game Theory

Game theory is concerned with situations where a set of players, each having a local utility function and set of actions (strategies), analyze strategies which maximize their own utilities [25, 79]. It is important to note that in this context, the global behavior arises as an "accident" of the individual players' choices, in that the players do not attempt either directly (take actions to that end) or indirectly (take actions that allow other players to take actions to that end) to maximize the global utility. In fact, the concept of global utility is defined as a a by-product of players' utilities, rather than be a desirable goal state in its own right.

In a game where each player analyzes the potential actions at a given time step, evaluates them on the basis of corresponding expected local utility and selects the most profitable strategy, it is important to study both convergence and equilibrium properties of the system [70, 192]. Although there are many types of equilibrium in a game, the most commonly used one was formalized by Nash [162].

In a Nash equilibrium, each player's strategy is the optimal response to the other player's strategies. In other words, Alternately, it is a state in the game where no player can improve its utility by changing its actions unilaterally. One of the reasons that the

Nash equilibrium is crucial in the analysis of games, is that it provides "consistent" predictions, i.e., if all parties predict that the game will converge to a Nash equilibrium, no one will benefit by changing strategies [79]. Note however, that a consistent prediction does not ensure an equilibrium point where the local utilities are maximized. The study of small perturbations around Nash equilibria from a stochastic dynamics perspective presents insight on how to select an equilibrium state when more than one are present [144].

The strategies that each player has at its disposal are also referred to as *pure strategies*, i.e., one takes a particular action every time one encounters a particular state. If on the other hand, a player chooses different strategies in a probabilistic manner, we refer to a *mixed strategy* game. We now state one of the most fundamental results in game theory: every finite game has a mixed strategy equilibrium, while it does not necessarily have a pure strategy equilibrium [79, 162]. The relevance of this result is in guaranteeing an equilibrium solution provided that agents are complex enough not to be restricted to always choose the same strategy in any one situation.

When agents play a game repeatedly, one can study the agent's performance over time and evaluate strategies accordingly. If agents learn to modify their strategies, one refers to repeated games with memory. If on the other hand agents have a fixed set of strategies but are either removed from the game or allowed to multiply according to their accumulated reward, one refers to *evolutionary game theory*. Within this framework one can study the long term effects of strategies such a cooperation and see if they arise naturally and if so, under what circumstances [8, 21, 66, 118, 165]; investigate the dependence of evolving strategies to the amount of information available to the agents [149]; study the effect of communication on the evolution of cooperation [150, 152]; and draw parallels with auctions and economic theory [98, 151].

3.2.5 El Farol Bar Problem

The "El Farol" bar problem and its variants provide a clean and simple testbed for investigating certain kinds of interactions among agents [4, 44]. In the original version of the problem, which arose in economics, at each time step (each "night"), each agent needs to decide whether to attend a bar. The goal of the agent in making this decision depends on the total attendance at the bar on that night. If the total attendence is below a preset capacity then the agent should have attended. Conversely, if the bar is overcrowded on the given night, then the agent should not attend. (Because of this structure, the bar problem with capacity set to 50% of the total number of agents is also known as the 'minority game'; each agent selects one of two groups at each time step, and those that are in the minority have made the right choice). The agents make their choices by predicting ahead of time whether the attendance on the current night will exceed the capacity and then taking the appropriate course of action.

What makes this problem particularly interesting is that it is impossible for all agents to be perfectly rational in the sense of all correctly predicting the attendance on any given night. This is because if most agents predict that the attendance will be low (and therefore decide to attend), the attendance will actually high, while if they predict the attendance will be high (and therefore decide not to attend) the attendance will be low. (In the language of game theory, this essentially amounts to the property that there are no pure strategy Nash equilibria [46, 226].) Alternatively, viewing the overall system as a COIN, it has a Prisoner's Dillema-like nature, in that rational behavior by all the individual agents thwarts the global goal of maximizing total enjoyment (defined as the sum of all agents' enjoyment and maximized when the bar is exactly at capacity).

This frustration effect is similar to what occurs in spin glasses in physics, and makes the bar problemclosely related to the physics of emergent behavior in distributed systems [45, 44, 229, 43]. Researchers have also studied the dynamics of the bar problem to investigate economic properties like competition, cooperation and collective behavior and especially their relationship to market efficiency [57, 184, 114].

3.3 Biologically Inspired Systems

Properly speaking, biological systems do not involve utility functions and searches across them with RL algorithms. However it has long been appreciated that there are many ways in which viewing biological systems as involving searches over such functions can lead to deeper understanding of them [189, 224]. Conversely, some have argued that the mechanism underlying biological systems can be used to help design search algorithms [99].[4]

These kinds of reasoning which relate utility functions and biological systems have traditionally focussed on the case of single a biological system operating in some external environment. If we extend this kind of reasoning, to a set of biological systems that are co-evolving with one another, then we have essentially arrived at biologically-based COINs. This section discusses some of how previous work in the literature bears on this relationship between COINs and biology.

3.3.1 Population Biology and Ecological Modeling

The fields of population biology and ecological modeling are concerned with the large-scale "emergent" processes that govern the systems that consist of many (relatively) simple entities interacting with one another [?]. As usually cast, the "simple entities" are members of one or more species, and the interactions are some mathematical abstraction of the process of natural selection as it occurs in biological systems (involving processes like genetic reproduction of various sorts, genotypy-phenotype mappings, inter and intra-species competitions for resources, etc.). Population Biology and ecological modeling in

[4] See [223, 141] though for some counter-arguments to the particular claims most commonly made in this regard.

this context addresses questions concerning the dynamics of the resultant ecosystem, and in particular how its long-term behavior depends on the details of the interactions between the constituent entities. Broadly construed, the paradigm of ecological modeling can even be broadened to study how natural selection and self-regulating feedback creates a stable planet-wide ecological environment—Gaia [135].

The underlying mathematical models of other fields can often be usefully modified to apply to the kinds of systems population biology is interested in [11]. Conversely, the underlying mathematical models of population biology and ecological modeling can be applied to other non-biological systems. In particular, those models shed light on social issues such as the emergence of language or culture, warfare, and economic competition [80, 67, 68]. They also can be used to investigate more abstract issues concerning the behavior of large complex systems with many interacting components [171, 90, 145, 164, 81].

Going a bit further afield, an approach that is related in spirt to ecological modeling is 'computational ecologies'. These are large distributed systems where each component of the system's acting (seemingly) independently results in complex global behavior. Those components are viewed as constituting an "ecology" in an abstract sense (although much of the mathematics is not derived from the traditional field of ecological modeling). In particular, one can investigate how the dynamics of the ecology is influenced by the information available to each component and how cooperation and communication among the components affects that dynamics [110, 108].

Although in some ways the most closely related to COINs of the current ecology-inspired research, the field of computational ecologies has some significant shortcomings if one tries to view it as a full science of COINs. In particular, it suffers from not being designed to solve the inverse problem of how to configure the system so as to arrive at a particular desired dynamics. This is a difficulty endemic to the general program of equating ecological modeling and population biology with the science of COINs. These fields are primarily concerned with the "forward problem" of determining the dynamics that arises from certain choices of the underlying system. Unless one's desired dynamics is sufficiently close to some dynamnics that was previously catalogued (during one's investigation of the forward prblem), one has very little information on how to set up the components and their interactions to achieve that desired dynamics. In addition, most of the work in these fields does not involve RL algorithms, and viewed as a context in which to design COINs suffers from a need for hand-tailoring, and potentially lack of robustness and scalability.

$\Longleftarrow$

Kagan: Should the following ref.'s really be here? [14, 191, 230, 62].

3.3.2 Swarm Intelligence

The field of 'swarm intelligence' is concerned with systems that are modelled after social insect colonies, so that the different components of the system are queen, worker, soldier, etc. It can be viewed as ecological modeling in which the individual entities have extremely limited computing capacity and/or action sets, and in which there are very few types of entities. The premise of the field is that the rich behavior of social insect colonies arises not from the sophistication of any individual entity in the colony, but from the interaction among those entities. The objective of current research is to uncover kinds of interactions among the entity types that lead to pre-specified behavior of some sort.

More speculatively, the study of social insect colonies may also provide insight into how to achieve learning in large distributed systems. This is because at the level of the individual insect in a colony, very little (or no) learning takes place. However across evolutionary time-scales the social insect species as a whole functions as if the various individual types in a colony had "learned" their specific functions. The "learning" is the direct result of natural selection. (See the discussion on this topic in the subsection on ecological modeling.)

Swarm intelligences have been used to adaptively allocate tasks in a mail company [28], solve the traveling salesman problem [60, 61] and route data efficiently in dynamic networks [27, 187, 202] among others. Despite this, such intelligences do not really constitute a general approach to designing COINs. There is no general framework for adapting swarm intelligences to maximize particular world utility functions. Accordingly, such intelligences generally need to be hand-tailored for each application. And after such tailoring, it is often quite a stretch to view the system as "biological" in any sense, rather than just a simple and *a priori* reasonable modification of some previously deployed system.

3.3.3 Artificial Life

The two main objective of Artifical Life, closely related to one another, are understanding the abstract functioning and especially the origin of terrestrial life, and creating organisms that can meaningfully be called "alive" [131].

The first objective involves formalizing and abstracting the mechanical processes underpinning terrestrial life. In particular, much of this work involves various degrees of abstraction of the process of self-replication [36, 197, 210]. Some of the more real-world-oriented work on this topic involves investigating how lipids assemble into more complex structures such as vesicles and membranes is one of the fundamental questions in the origin of life [59, 168, 64]. Many computer models have been proposed to simulate this process, though most suffer from overly simplifying the molecular morphology. **Kagan: We need some Pohorille/New ref.'s here.**

More generally, work concerned with the origin of life can constitute an investigation of the functional self-organization that gives rise to life [146]. In this regard, an important early work on functional self-organization is the *lambda calculus*, which provides an elegant framework (recursively defined functions, lack of distinction between object and function, lack of architectural restrictions) for studying computational systems [50]. This framework can be used to develop an artificial chemistry "function gas" that displays complex cooperative properties [74].

The second objective of the field of Artificial Life is less concerned with understanding the details of terrestrial life per se than of using terrestrial life as inspiration for how to design living systems. For example, motivated by the existence (and persistence) of computer viruses, several workers have tried to design an immune system for computers that will develop "antibodies" and handle viruses both more rapidly and more efficiently than other algorithms [119, **?**]. More enerally, because we only have one sampling point (life on Earth), it is very difficult to precisely formulate the process by which life emerged By creating an artificial world inside a computer however, it is possible to study far more general forms of life [175, 176, 177]. See also [222] where the argument is presented that the richest way of approaching the issue of defining "life" is phenomenologically, in terms of self-*dis*similar scaling properties of the system.

3.3.4 Training cellular automata with genetic algorithms

Cellular automata can be viewed as digital abstractions of physical gases [**?**]. Formally, they are discrete-time recurrent neural nets where the neurons live on a grid, each neuron has a finite number of potential states, and inter-neuron connections are (usually) purely local. (See below for a dicussion of recurrent neural nets.) So the state update rule of each neuron is fixed and local, the next state of a neuron being a function of the current states of it and of its neighboring elements.

The state update rule of (all the neurons making up) any particular cellular automaton specifies the mapping taking the initial configuration of the states of all of its neurons to the final, equilibrium (perhaps strange) attractor configuration of all those neurons. So consider the situation where we have a desired such mapping, and want to know an update rule that induces that mapping. This is a search problem, and can be viewed as similar to the inverse problem of how to design a COIN to achieve a pre-specified global goal, albeit a "COIN" whose nodal elements do not use RL algorithms.

Genetic algorithms are a special kind of search algorithm, based on analogy with the biological process of natural selection via recombination and mutation of a genome [**?**]. There is no formal theory justifying genetic algorithms as search algorithms [140, 223] and very few empirical comparisons with other search techniques that mightjustify their use. Nonetheless, genetic algorithms (and 'evolutionary computation' in general) have been studied quite extensively. In particular, they have been used to (try to) solve the inverse problem of finding update rules for a cellular automaton that induce a pre-specified mapping from its initial configuration to its attractor configuration. To date, they have used this way only for extremely simple configuration mappings, mappings which can be trivially learned by other kinds of systems. Despite the simplicity of these mappings, the use of genetic algorithms to try to train cellular automata to exhibit them has achieved little success [155, 154, 53, 55].

3.4 Physics-Based Systems

3.4.1 Statistical Physics

Equilibrium statistical physics is concerned with the stable state character of large numbers of very simple physical objects, interacting according to well-specified local deterministic laws, with probabilistic noise processes superimposed [?, ?]. Typically there is no sense in which such systems can be said to have centralized control, since all particles contribute comparably to the overall dynamics.

Aside from mesoscopic statistical physics, the numbers of particles considered are usually on the order of 10^{23}, and the particles themselves are extraordinarily simple, typically having only a few degrees of freedom. Moreover, the noise processes usually considered are highly restricted, being those that are formed by "baths", of heat, particles, and the like. Similarly, almost all of the field restricts itself to deterministic laws that are readily encapsulated in Hamilton's equations (Schrodinger's equation and its field-theoretic variants for quantum statistical physics). In fact, much of equilibrium statistical physics isn't even concerned with the dynamic laws by themselves (as for example is stochastic Markov processes). Rather it is concerned with invariants of those laws (e.g., energy), invariants that relate the states of all of the particles. Trivially then, deterministic laws without such readily-discoverable invariants are outside of the purview of much of statistical physics.

One potential use of statistical physics for COINs involves taking the systems that statistical physics analyzes, especially those analyzed in its condensed matter variant (e.g., spin glasses [?]), as simplified models of a class of COINs. This approach is used in some of the analysis of the Bar problem (see above). It is used more overtly in (for example) the work of Galam [82], in which the equilibrium coalitions of a set of "countries" are modeled in terms of spin glasses. This approach cannot provide a general COIN framework though. In addition to the caveats listed above, this is due to its not providing a general solution to inverse problems and its lack of RL algorithms.[5]

Another contribution that statistical physics can make is with the mathematical techniques it has developed for its own purposes, like mean field theory, self-averaging ap-

[5]In regard to the latter point however, it's interesting to speculate about recasting statistical physics as a COIN, by having each of the particles in the physical system run an RL algorithm that perfectly optimizes the "utility function" of its Lagangian, given the "actions" of the other particles. In this perspective, many-particle physical systems are multi-stage games that are at Nash equilibrium in each stage.

proximations, phase transitions, Monte Carlo techniques, the replica trick, and tools to analyze the thermodynamic limit in which the number of particles goes to infinite. Although such techniques have not yet been applied to COINs, the have been successfully applied to related fields. This is exemplified by the use of the replica trick to analyze two-player zero-sum games with random payoff matrices in the thermodynamic limit of the number of strategies in [22]. Other examples are the numeric investigation of iterated prisoner's dilemma played on a lattice [206], the analysis of stochastic games by expressing of deviation from rationality in the form of a "heat bath" [144], and the use of topological entropy to quantify the complexity of a voting system studied in [147].

Other work in the statistical physics literature is formally identical to that in other fields, but presents it from a novel perspective. A good example of this is [196] which analyzes use of a single simple proportional RL algorithm for control of a spatially extended system. (All without a single mention of the field of reinforcement learning.)

3.4.2 Action Extremization

Much of the theory of physics can be cast as solving for the extremization of an actional, which is a functional of the worldline of an entire (potentially many-component) system across all time. The solution to that extremization problem constitutes the actual worldline followed by the system. In this way the calculus of variations can be used to solve for the worldline of a dynamic system. As an example, simple Newtonian dynamics can be cast as solving for the worldline of the system that extremizes a quantity called "the Lagrangian", which is a function of that worldline and of certain parameters (e.g., the "potential energy") governing the system at hand. In this instance, the calculus of variations simply results in Newton's laws.

If we take the dynamic system to be a COIN, we are assured that its worldline automatically optimizes a "global goal" consisting of the value of the associated actional. If we change physical aspects of the system that determine the functional form of the actional (e.g., change the system's potential energy function), then we change the global goal, and we are assured that our COIN optimizes that new global goal.

The challenge in exploiting this to solve the inverse problem of how to design physical COINs is in translating an arbitrary provided global goal for the COIN into a parameterized actional. Note that that actional must govern the dynamics of the physical COIN, and the parameters of the actional must be physical variables in the COIN, variables whose values we can modify.

3.4.3 Active Walker Models

The field of active walker models [17, 93, 92] is concerned with modeling "walkers" (be they human walkers or instead simple physical objects) crossing fields along trajectories, where those trajectories are a function of several factors, including in particular the trails already worn into the field. Often the kind of trajectories considered are those that can be cast as solutions to actional extremization problems so that the walkers can be explicitly viewed as agents optimizing a private utility.

One of the primary concerns with the field of active walker models is how the trails worn in the field change with time to reach a final equilibrium state. The problem of how to design the cement pathways in the field (and other physical features of the field) so that the final paths actually followed by the walkers will have certain desirable characteristics is then one of solving for parameters of the actional that will result in the desired worldline. This is a special instance of the inverse problem of how to design a COIN.

Using active walker models this way to design COINs, like action extremization in general, probably has limited applicability. Also, it is not clear how robust such a design approach might be, or whether it would be scalable and exempt from the need for hand-tailoring.

3.5 Other Related Subjects

This subsection presents a "catch-all" of other fields that have little in common with one another except that they bear some relation to COINs.

3.5.1 Stochastic Fields

An extremely well-researched body of work concerns the mathematical and numeric behavior of systems for which the probability distribution over possible future states conditioned on preceding states is explicitly provided. This work involves many aspects of Monte Carlo numerical algorithms [**?**], all of Markov Chains [**?**], and especially Markov fields, a topic that encompasses the Chapman-Kolmogorov equations [83] and its variants: Liouville's equation, the Fokker-Plank equation, and the Detailed-balance equation in particular. Non-linear dynamics is also related to this body of work (see the synopsis of iterated function systems below and the synopsis of cellular automata above), as is Markov competitive decision processes (see the synopsis of game theory above).

Formally, one can cast the problem of designing a COIN as how to fix each of the conditional transition probability distributions of the individual elements of a stochastic field so that the aggregate behavior of the overall system is of a desired form.[6] Unfortunately, almost all that is known in this area instead concerns the forward problem, of

[6]In contrast, in the field of Markov decision processes, discussed in [42], the full system may be a Markov field, but the system designer only sets the conditional transition probability distribution of a few of the field elements at most, to the appropriate "decision rules". Unfortunately, it is hard to imagine how to use the results of this field to design COINs because of major scaling problems. Any decision process must accurately model likely future modifications to its own behavior — often an extremely daunting task [141]. What's worse, if multiple such decision processes are running concurrently in the system, each such process must also model the others, in their full complexity.

inferring aggregate behavior from a provided set of conditional distributions. Although such knowledge provides many "bits and pieces" of information about how to tackle the inverse problem, those pieces collectively cover only a very small subset of the entire space of tasks we might want the COIN to perform. In particular, they tell us very little about the case where the conditional distribution encapsulates RL algorithms.

Put [136], somewhere. $\Longleftarrow$

3.5.2 Iterated Function Systems

The technique of iterated function systems [15, **?**] grew out of the field of nonlinear dynamics [**?**, **?**]. In such systems a function is repeatedly and recursively applied to itself. The most famous example is the logistic map, $x_{n+1} = rx_n(1 - x_n)$ for some r between 0 and 4 (so that x stays between 0 and 1). More generally the function along with its arguments can be vector-valued. In particular, we can construct such functions out of affine transfromations of points in a Euclidean plane.

Iterated functions systems have been applied to image data. In this case the successive iteration of the function generically generates a fractal, one whose precise character is determined by the initial iteration-1 image. Since fractals are ubiquitous in natural images, a natural idea is to to try to encode natural images as sets of iterated function systems spread across the plane, thereby potentially garnering significant image compression. The trick is to manage the inverse step of starting with the image to be compressed, and determining what iteration-1 image(s) and iterating function(s) will generate an accurate approximation of that image.

In the language of nonlinear dynamics, we have a dynamic system that consists of a set of iterating functions, together with a desired attractor (the image to be compressed). Our goal is to determine what values to set certain parameters of our dynamic system to so that the system will have that desired attractor. The potential relationship with COINs arises from this inverse nature of the problem tackled by iterated function systems. If the goal for a COIN can be cast as its relaxing to a particular attractor, and if the distributed computational elements are isomorphic to iterated functions, then the tricks used in iterated functions theory could be of use.

Although the techniques of iterated function systems might prove of use in designing COINs, they are unlikely to serve as a generally applicable approach to designing COINs. In addition, they do not involve RL algorithms, and often involve extensive hand-tuning.

3.5.3 Recurrent Neural Nets

A recurrent neural net consists of a finite set of "neurons" each of which has a real-valued state at each moment in time. Each neuron's state is updated at each moment in time based on its current state and that of some of the other neurons in the system. The topology of such dependencies constitute the "inter-neuronal connections" of the net, and the associated parameters are often called the "weights" of the net. The dynamics can be either discrete or continuous (i.e., given by difference or differential equations).

Recurrent nets have been investigated for many purposes [41, 103, 84, 169, 227]. One of the more famous of these is associative memories. The idea is that given a pre-specified pattern for the (states of the neurons in the) net, there may exist inter-neuronal weights which result in a basin of attraction focussed on that pattern. If this is the case, then the net is equivalent to an associative memory, in that a complete pre-specified pattern across all neurons will emerge under the net's dynamics from any initial pattern that partially matches the full pre-specified pattern. In practice, one wishes the net to simultaneously possess many such pre-specified associative memories. There are many schemes for "training" a recurrent net to have this property, including schemes based on spin glasses [100, 101, 102] and schemes based on gradient descent [179].

As can the fields of cellular automata and iterated function systems, the field of recurrent neural nets can be viewed as concerning certain variants of COINs. Also like those other fields though, recurrent neural nets has shortcomings if one tries to view it as a general approach to a science of COINs. In particular, recurrent neural nets do not involve RL algorithms, and training them often suffers from scaling problmes. More generally, in practice they can be hard to train well without hand-tailoring.

3.5.4 Network Theory

Packet routing in a data network [23, 104, 198, 212] presents a particularly interesting domain for the investigation of COINs. In particular, with such routing:
(i) the problem is inherently distributed;
(ii) for all but the most trivial networks it is impossible to employ global control ;
(iii) the routers have only access to local information (routing tables);
(iv) it constitutes a relatively clean and easily modified
experimental testbed; and
(v) there are potentially major bottlenecks induced by 'greedy'
behavior on the part of the individual routers, which behavior
constitutes a readily investigated instance of the TOC.

Many of the approaches to packet routing incorporate a variant on RL [34, 137, 39, 48, 143]. Q–routing is perhaps the best known such approach and is based on routers using reinforcement learning to select the best path [34]. Although generally successful, Q–routing is not a general scheme for inverting a global task. This is even true if one restricts attention to the problem of routing in data networks — there exists a global task in such problems, but that task is directly used to construct the algorithm.

A particular version of the general packet routing problem that is acquiring increased attention is the Quality of Service (QoS) problem, where different communication packets (voice, video, data) share the same bandwidth resource but have widely varying importances both to the user and (via revenue) to the bandwidth provider. Determining which packet has precedence over which other packets in such cases is not only based on priority in arrival time but more generally on the potential effects on the income of the bandwidth provider. In this context, RL algorithms have been used to determine routing policy, control call admission and maximize revenue by allocation the available bandwidth efficiently [39, 143].

Many researchers have exploited the noncooperative game theoretic understanding of the TOC in order to explain the bottleneck character of empirical data networks behavior and suggest potential alternatives to current routing schemes [132, 124, 166, 167, 193, 130, 63]. Closely related is work on various "pricing"-based resource allocation strategies in congestable data networks [139]. This work is at least partially based upon current understanding of pricing in toll lanes, and traffic flow in general (see below). All of these approaches are particularly of interest when combined with the RL-based schemes mentioned just above. Due to these factors, much of the current research on a general framework for COINs is directed toward the packet-routing domain (see next section).

3.5.5 Traffic Theory

Traffic congestion typifies the TOC public good problem: everyone wants to use the same resource, and all parties greedily trying to optimize their use of that resource not only worsens global behavior, but also worsens *their own* personal utility (e.g., if everyone disobeys traffic lights, everyone gets stuck in traffic jams). Indeed, in the well-known Braess' paradox [16], keeping everything else constant — including the number and destinations of the drivers — but opening a new traffic path can *increase* everyone's time to get to their destination. (Viewing the overall system as in instance of the Prisoner's dilemma, this paradox in essence arises through the creation of a novel 'defect-defect' option for the overall system.) Greedy behavior on the part of individuals also results in very rich global dynamic patterns, such as stop and go waves and clusters [94, 95].

Much of traffic theory employs and investigates tools that have previously been applied in statistical physics [170, 121, 122, 174, 94] (see subsection above). In particular, the spontaneous formation of traffic jams provides a rich testbed for studying the emergence of complex activity from seemingly chaotic states [96, 94]. Furthermore, the dynamics of traffic flow is particular amenable to the application and testing of many novel numerical methods in a controlled environment [13, 24, 188]. Many experimental studies have confirmed the usefulness of applying insights gleaned from such work to real world traffic scenarios [161, 160, 94].

3.5.6 Topics from further afield

Finally, there are a number of other fields that, while either still nascent or not extremely closely related to COINs, are of interest in COIN design:

Amorphous computing: Amorphous computing grew out of the idea of replacing traditional computer design, with its requirements for high reliability of the components of the computer, with a novel approach in which widespread unreliability of those components would not interfere with the computation [1]. Some of its more speculative aspects are concerned with "how to program" a massively distributed, noisy system of components which may consist in part of biochemical and/or biomechanical components [123, 216]. Work here has tended to focus on schemes for how to robustly induce desired geometric dynamics across the physical body of the amorphous computer — issue that are closely related to morphogenesis, and thereby lend credence to the idea that biochemical components are a promising approach. Especially in its limit of computers with very small constituent components, amorphous computing also is closely related to the fields of nanotechnology [**?**] and control of smart matter (see below).

Control of smart matter:. As the prospect of nanotechnology-driven mechanical systems gets more concrete, the daunting problem of how to robustly control, power, and sustain protean systems made up of extremely large sets of nano-scale devices looms more important [88, 87, 97]. If this problem were to be solved one would in essence have "smart matter". For example, one would be able to "paint" an airplane wing with such matter and have it improve drag and lift properties significantly.

Morphogenesis: How does a leopard embryo get its spots, or a zebra embryo its stripes? More generally, what are the processes underlying morphogenesis, in which a body plan develops among a growing set of initially undifferentiated cells? These questions, related to control of the dynamics of chemical reaction waves, are essentially special cases of the more general question of how ontogeny works, of how the genotype-phenotype mapping is carried out in development. The answers involve homeobox (as well as many other) genes [**?**, **?**]. Under the presumption that the functioning of such genes is at least in part designed to facilitate genetic changes that increase a species' fitness, that functioning facilitates solution of the inverse problem, of finding small-scale changes (to DNA) that will result in "desired" large scale effects (to body plan) when propagated across a growing distributed system. **Kagan: Get some standard reference on homeobox genes and ontogeny, and chemical reactions governed by Turing's equation.**

Self Organizing systems The concept of self-organization and self-organized criticality [12] was originally developed to help understand why many distributed physical systems are attracted to critical states that possess long-range dynamic correlations in the large-scale characteristics of the system. It provides a powerful framework for analyzing both biological and economic systems. For example, natural selection (particularly punctuated equilibrium [86]) can be likened to self-organizing dynamical system, and some have argued it shares many the properties (e.g., scale invariance) of such systems [56]. Similarly, one can view the economic order that results from the actions of human agents as a case of self-organization [58]. The relationship between complexity and self-organization is a particularly important one, in that it provides the potential laws that allow order to arise from chaos [117].

Small worlds (6 Degrees of Separation): In many distributed systems where each component can interact with a small number of "neighbors", an important problem is how to propagate information across the system quickly and with minimal overhead. On the one extreme the neighborhood topology of such systems can exist on a completely regular grid-like structure. On the other, the topology can be totally random. In either case, certain nodes may be effectively 'cut-off' from other nodes if the information pathways between them are too long. Recent work has investigated "small worlds" networks (sometimes called 6 degrees of separation) in which underlying grid-like topologies are "doped" with a scattering of long-range, random connections. It turns out that very little such doping is necessary to allow for the system to effectively circumvent the information propogation problem [148, 215].

Control theory: Adaptive control [**?**], and in particular adaptive control involving locally weighted RL algorithms [5, 157], constitute a broadly applicable framework for controlling small, potentially inexactly modeled systems. Augmented by techniques in the control of chaotic systems [49, **?**], they constitute a very successful way of solving the "inverse problem" for such systems. Unfortunately, it is not clear how one could even attempt to scale such techniques up to the massively distributed systems of interest in COINs. The next section discusses in detail some of the underlying reasons why the purely model-based versions of these approaches are inappropriate as a framework for COINs.

[**?**]

4 A FRAMEWORK DESIGNED FOR COINs

Summarizing the discussion to this point, it is hard to see how any already extant scientific field can be modified to encompass systems meeting all of the requirements of COINs listed at the beginning of Section 3. This is not too surprising, since none of those fields were explicitly designed to analyze COINs. This section first motivates in general terms a framework that is explicitly designed for analyzing COINs. It then presents the formal nomenclature of that framework. This is followed by deriving some of the central theorems of that framework. Finally, we present experiments that illustrate the power the framework provides for ensuring large world utility in a COIN.

Unfortunately, for reasons of space, the discussion here is abbreviated and laconic. A much more detailed discussion, including intuitive arguemnts, proofs and fully formal definitions of the concepts discussion in this section, can be found in [219].

4.1 Problems with a model-based approach

What mathematics might one employ to understand and design COINs? Perhaps the most natural approach, related to the stochastic fields work reviewed above, involves the following three steps:

1) First one constructs a complete stochastic model of the COIN's dynamics, a model parameterized by a vector θ. As an example, θ could fix the utility functions of the individual agents of the COIN, aspects of their RL algorithms, which agents communicate with each other and how, etc.

2) Next we solve for the function $f(\theta)$ which maps the parameters of the model to the resulting stochastic dynamics.

3) Cast our goal for the system as a whole as achieving a high expected value of some "world utility". Then as our final step we would have to solve the inverse problem: we would have to search for a θ which, via f, results in a high value of E(world utility $\mid \theta$).

Let's examine in turn some of the challenges each of these three steps entrain:

I) We are primarily interested in very large, very complex systems, which are noisy, faulty, and often operate in a non-stationary environment. Moreover, our "very complex system" consists of many RL algorithms, all potentially quite complicated, all running simultaneously. Clearly coming up with a model that captures the dynamics of all of this in an accurate manner will often be extraordinarily difficult. Moreover, unfortunately, often the level of versimiltude required of the model will be quite high. For example, unless the modeling of the faulty aspects of the system were quite accurate, the model would likely be "brittle", and overly sensitive to which elements of the COIN were and were not operating properly at any given time.

II) Even for models much simpler than the ones called for in (I), solving explicitly for the function f can be extremely difficult. For example, much of Markov Chain theory is an attempt to broadly characterize such mappings. However as a practical matter, usually it can only produce potentially useful characterizations when the underlying models are quite inaccurate simplifications of the kinds of models produced in step (I).

III) Even if one can write down an f, solving the associated inverse problem is often impossible in practice.

IV) In addition to these difficulties, there is a more general problem with the model-based approach. We wish to perform our analysis on a "high level". Our thesis is that due to the robust and adaptive nature of the individual agents' RL algorithms, there will be very broad, easily identifiable regions of θ space all of which result in excellent E(world utility $\mid \theta$), and that these regions will not depend on the precise learning algorithms used to achieve the low-level tasks (cf. the list at the beginning of Section 3). To fully capitalize on this one would want to be able to slot in and out different learning algorithms for achieving the low-level tasks without having to redo our entire analysis each time. However in general this would be possible with a model-based analysis only for very carefully designed models (if at all). The problem is that the result of step (3), the solution to the inverse problem, would have to concern aspects of the COIN that are (at least approximately) invariant with respect to the precise low-level learning algorithms used. Coming up with a model that has this property while still avoiding problems (I-III) is usually an extremely daunting challenge.

Fortunately, there is an alternative approach which avoids modeling and its associated difficulties. We call any framework based on this alternative a **descriptive framework**. In such a framework one identifies certain **salient characteristics** of COINs, which are characteristics that one strongly expects to find in COINs that have large world utility. Under this expectation, one assumes that if a COIN is explicitly modified to have the salient characteristics, perhaps in response to observations of its run-time behavior, then its world utility will benefit. If those salient characteristics are (relatively) easy to induce in a COIN, then this assumption provides a ready way to cause that COIN to have large world utility. If in addition the salient characteristics can be induced with little or no modeling (e.g., via heuristics that aren't rigorously and formally justified), then the descriptive framework can be used to improve world utility without recourse to detailed modeling.

4.2 Nomenclature

There exist many ways one might try to design a descriptive framework. In this subsection we present nomenclature needed for a (very) cursory overview of one of them. (See [219] for a more detailed exposition, including formal proofs.)

4.2.1 Preliminary Definitions

1) We refer to an RL algorithm by which an individual component of the COIN modifies its behavior as a **microlearning** algorithm. We refer to the initial construction of the COIN, potentially based upon salient characteristics, as the COIN **initialization**. We use the phrase **macrolearning** to refer to externally imposed run-time modifications to the COIN which are based on statistical inference concerning salient characteristics of the running COIN.

2) For convenience, we take time t to be discrete and confined to the integers. When referring to COIN initialization, we implicitly have a lower bound on t, which without loss of generality we take to be ≤ 0.

3) All variables that have any effect on the COIN are identified as components of Euclidean-vector-valued **states** of various discrete **nodes**. So for example, if our COIN consists in part of an "agent" running a set of microlearning algorithm, the precise configuration of that agent at any time t, including all variables in its learning algorithm, all externally visible actions, internal parameters, values observed by its probes of the surrounding environment, etc., all constitute the state vector of a node representing that

agent. We define $\underline{\zeta}_{\eta,t} \in \underline{\mathbf{Z}}_{\eta,t}$ to be the Euclidean vector giving the state of node η at time t.

4) For notational convenience, we define $\underline{\zeta}_{,t} \in \underline{\mathbf{Z}}_{,t}$ to be the vector of the states of all nodes at time t; $\underline{\zeta}_{\hat{\eta},t} \in \underline{\mathbf{Z}}_{\hat{\eta},t}$ to be the vector of the states of all nodes other than η at time t; and $\underline{\zeta} \equiv \underline{\zeta}_{,} \in \underline{\mathbf{Z}}$ to be the entire vector of the states of all nodes at all times. $\underline{\mathbf{Z}}$ is infinite-dimensional in general, and usually assumed to be a Hilbert space. Also for notational convenience, we define gradients using ∂-shorthand. So for example, $\partial_{\underline{\zeta}_{,t}} F(\underline{\zeta}_{,})$ is the vector of the partial derivative of $F(\underline{\zeta}_{,})$ with respect to the components of $\underline{\zeta}_{,t}$.

5) We assume the universe in which our COIN operates is completely deterministic. (E.g., it obeys classical physics, or quantum physics with no collapse of the wave packet.) Formally, we do this by bundling all variables we're not directly considering — but which nonetheless affect the dynamics of the system — as components of some catch-all **environment node**. Then we stipulate that for all $t, t' > t$, $\underline{\zeta}_{,t}$ sets $\underline{\zeta}_{,t'}$ uniquely.

We will often be concerned with physical systems obeying entropy-driven contractive (i.e., error-correcting) dynamic processes. Formally, rather than encapsulate such processes by having $\underline{\mathbf{Z}}$ be the phase space of every particle in the system, we will usually have $\underline{\mathbf{Z}}$ consist of variables existing at a larger scale (e.g., thermodynamic variables in the thermodynamic limit). This in turn means that although the dynamics of our system is deterministic, it need not be invertible.

We express the dynamics of our system by writing $\underline{\zeta}_{,t' \geq t} = C(\underline{\zeta}_{,t})$. (In this paper there will be no need to be more precise and specify the precise dependency of $C(.)$ on t and/or t'.) We define $\{C\}$ to be a set of constraint equations enforcing that dynamics, and more generally fixing the manifold C of vectors $\underline{\zeta} \in \underline{\mathbf{Z}}$ that we consider to be 'allowed'. So C is a subset of the set of all $\underline{\zeta} \in \underline{\mathbf{Z}}$ that are consistent with the deterministic laws governing the COIN, i.e., that obey $\underline{\zeta}_{,t' \geq t} = C(\underline{\zeta}_{,t}) \; \forall \; t, t'$. We generalize this notation in the obvious way, so that (for example) $C_{,t \geq t_0}$ is the manifold consisting of all vectors $\underline{\zeta}_{,t \geq t_0} \in \underline{\mathbf{Z}}_{,t \geq t_0}$ that are projections of a vector in C.

Note that $C_{,t \geq t_0}$ is parameterized by $\underline{\zeta}_{,t_0}$, due to determinism. Note also that whereas $C(.)$ is defined for any argument of the form $\underline{\zeta}_{,t} \in \underline{\mathbf{Z}}_{,\mathbf{t}}$ for some t (i.e., we can evolve any point forward in time), in general not all $\underline{\zeta}_{,t} \in \underline{\mathbf{Z}}_{,\mathbf{t}}$ lie in $C_{,t}$. In particular, we as COIN designers can impose extra restrictions on the possible states of the system beyond its need to obey the relevant dynamical laws of physics.

In this approach, all behavior across time is pre-fixed. The COIN is a single fixed worldline through $\underline{\mathbf{Z}}$, with no "unfolding of the future" as the die underlying a stochastic dynamics get cast. This is consistent with the fact that we want the formalism to be purely descriptive, relating different properties of any single, fixed COIN's history. We will often informally refer to "changing a node's state at a particular time", or to a microlearner's "choosing from a set of options", and the like. Formally, in all such phrases we are really comparing different worldlines, with the indicated modification distinguishing those worldlines.

The deterministic nature of our framework does not preclude our incorporating probabilistic elements into the framework. Exactly as in statistical physics, a stochastic nature can be superimposed on top of our space of deterministic worldlines. Whereas the deterministic analysis presented here is related to game-theoretic structures like Nash equilibria, such a stochastic extension is more related to structures like correlated equilibria [7].

Formally, there is a lot of freedom in setting the boundary between what we call "the COIN", whose dynamics is determined by C, and what we call "macrolearning", which constitutes perturbations to the COIN instigated from "outside the COIN", and which therefore is *not* reflected in C. As an example, in much of this paper, we have clearly specified microlearners which are provided private utility functions that they are trying to maximize. In such cases usually we will implicitly take C to be the dynamics of the system, microlearning and all, *for fixed private utilities* that are specified in $\underline{\zeta}$. Macrolearning overrides C, and in this situation it refers (for example) to any statistical inference process that modifies the private utilities at run-time to (try to) induce the desired salient characteristics. Since C does not reflect such macrolearning, when trying to ascertain C based on empirical observation (as for example when determining how best to modify the private utilities), we have to take care to distinguish which part of the system's observed dynamics is due to C and which part instead reflects externally imposed modifications to the private utilities.

More generally though, other boundaries between the COIN and macrolearning-based perturbations to it are possible. For these alternatives, we must scrutinize different aspects of the COIN's dynamics to infer C. Whatever the boundary, the mathematics of the descriptive framework, including the mathematics concerning the salient characteristics, is restricted to a system evolving according to C, and explicitly does not account for macrolearning. This is why the strategy of trying to improve world utility by using macrolearning to try to induce salient characteristics is ultimately based on an assumption rather than a proof.

6) We are provided with some Von Neumann **world utility** $G : \underline{\mathbf{Z}} \rightarrow \mathcal{R}$ that ranks the various conceivable worldlines of the COIN. We define **personal utilities** $g_\eta : \underline{\mathbf{Z}} \rightarrow \mathcal{R}$ similarly.

These utility definitions are very broad. In particular, they do not require casting of the utilities as discounted sums. Note also that our utilities are not indexed by t. Again reflecting the descriptive, worldline character of the formalism, we simply assign a single value to an entire worldline, implicitly assuming that one can always say which of two candidate worldlines are preferable. So given some "present time" t_0, issues like which of two "potential futures" $\underline{\zeta}_{,t>t_0}, \underline{\zeta}'_{,t>t_0}$ is preferable are resolved by evaluating the relevant utility at two associated points $\underline{\zeta}$ and $\underline{\zeta}'$, where the $t > t_0$ components of those points are the futures indicated, and the two points share the same (usually implicit) $t \leq t_0$ "past" components. This allows us to sidestep formal problems that can occur with general (i.e., not necessarly discounted sum) time-indexed utilities, problems like having what's optimal at one moment in time conflict with what's optimal at other moments in time.[7]

As mentioned above, there may be variables in each node's state which, under one particular interpretation, represent the "utility functions" that the associated microlearner's computer program is trying to extremize. When there are such components of $\underline{\zeta}$, we refer to the utilities they represent as **private utilities**. However even when there are private utilities, formally we allow the personal utilities to differ from them. The personal utility functions $\{g_\eta\}$ do not exist "inside the COIN"; they are not specified by components of $\underline{\zeta}$. This separating of the private utilities from the $\{g_\eta\}$ will allow us to avoid the teleological problem that one may not always be able to explicitly identify "the" private utility function reflected in $\underline{\zeta}$ such that a particular computational device can be said to be a microlearner "trying to increase the value of its private utility". To the degree that we can couch the theorems purely in terms of personal rather than private utilities, we will have successfully adopted a purely behaviorist approach, without any need to interpret what a computational device is "trying to do".

Despite this formal distinction though, often we will implicitly have in mind deploying the personal utilities onto the microlearners as their private utilities, in which case the terms can usually be used interchangeably. The context should make it clear when this is the case.

4.2.2 Intelligence

We will need a measure of the performance of an arbitrary worldline $\underline{\zeta}$ for an arbitrary utility function under arbitrary dynamic laws C. Such a measure is a mapping from three arguments to $\mathbf{R}$. Having such a measure will allow us to quantify how well the entire system performs in terms of G. It will also allow us to quantify how well each microlearner performs in purely behavioral terms, in terms of its personal utility. (In our behaviorist approach, we do not try to make specious distinctions between whether a microlearer performs well due to its "innate sophistication", or rather "by sheer luck" — all that matters is how effective its behavior is.) This behaviorism in turn will allow us to avoid having private utilities explicitly arise in our theorems (although they still arise frequently in pedagogical discussion). Even when private utilities exist, there will be no formal need to explicitly identify some components of $\underline{\zeta}$ as such utilities. Assuming a node's microlearner is competent, the fact that it is trying to optimize some particular private utility U will be manifested in our performance measure's having a large value at $\underline{\zeta}$ for C for that utility U.

[7] Such conflicts can be especially troublesome when they interfere with our defining what we mean by an "optimal" action by a node η at a particular time t. Their ability to interfere in this way is due to the fact that the effects of an action by η depend on future behavior of η, which (if it too is to be optimal) will depend on *its* future, etc., all in a potentially non-self-consistent infinite regress.

The problem of how to formally define such a performance measure is essentially equivalent to the problem of how to quantify bounded rationality in game theory. Some of the relevant work in game theory is concerned with refinements of equilibra, and adopts a strongly teleological perspective on rationality [?]). In general, such work is only narrowly applicable, to those situations where the rationality is bounded due to the precise causal mechanisms investigated in that work. Most of the other game-theoretic work first models (!) the microlearner, as some extremely simply computational device (e.g., a deterministic finite automaton (DFA)). One then assumes that the microlearner performs perfectly for that device, so that one can measure that learner's performance in terms of some computational capacity measure of the model (e.g., for a DFA, the number of states of that DFA) [78, 163, 183]. However if taken as renditions of real-world computer-based microlearners (never mind human microlearners!), the models in this approach are often extremely abstracted, with many important characteristics of the real learners absent or distorted. In addition, there is little reason to believe that any results arising from this approach would not be highly dependent on the model choice and on the associated representation of computational capacity. Yet another disadvantage is that this approach concentrates on perfect, fully rational behavior of the microlearners.

We would prefer a less model-dependent approach, one based soley on the utility function at hand, $\underline{\zeta}$, and C. Now we don't want our performance measure to be a "raw" utility value like $g_\eta(\underline{\zeta})$, since that is not invariant with respect to monotonic transformations of g_η. Similarly, we don't want to penalize the microlearner for not achieving a certain utility value if that value was impossible to achieve due to C and the actions of other nodes. A natural way to address these concerns is to generalize the game-theoretic concept of "best-response strategy" and consider the problem of how well η performs *given the actions of the other nodes.* Such a measure would compare the possible states of η at some particular time, which without loss of generality we can take to be 0, to the actual state $\underline{\zeta}_{\eta,0}$. In other words, we would compare the utility of the actual worldline $\underline{\zeta}$ to those of a set of alternative worldlines $\underline{\zeta}'$, where $\underline{\zeta}_{\hat{\eta},0} = \underline{\zeta}'_{\hat{\eta},0}$, and use those comparisons to quantify the quality of η's performance.

Now we're only concerned with comparing the effects of replacing $\underline{\zeta}$ with $\underline{\zeta}'$ on *future* contributions to the utility. But if we allow arbitrary $\underline{\zeta}'_{,t<0}$, then in and of themselves the difference between those past components of $\underline{\zeta}'$ and those of $\underline{\zeta}$ can modify the value of the utility, regardless of the effects of any difference in the future components. Our presumption is that for all COINs of interest we can avoid this conundrum by restricting attention to those $\underline{\zeta}'$ where $\underline{\zeta}'_{,t<0}$ differs from $\underline{\zeta}_{,t<0}$ only in the internal parameters of η's microlearner, differences that only at times $t \geq 0$ manifest themselves in a form the utility is concerned with. (In game-theoretic terms, such "internal parameters" encode full extensive form strategies, and we only consider changes to the vertices at or below the $t = 0$ level in the tree of an extensive-form strategy.) Under this presumption, without violating C, we're able to pose the question, "if we change the state of η at time 0 in such-and-such a way, leaving everything else of interest at that time unchanged, what are the ramifications on the utility?"

However we don't want to restrict the computational algorithms that can run on a node to those that have a clearly pre-specified set of "internal parameters" and the like. So instead, we formalize our presumption behaviorally. Since changing the internal parameters doesn't affect the $t < 0$ components of $\underline{\zeta}_{\eta,}$ *that the utility is concerned with*, and since we are only concerned with changes to $\underline{\zeta}$ that affect the utility, we simply elect to not change the $t < 0$ values of the internal parameters of $\underline{\zeta}_{\eta,}$ at all. In other words, we leave $\underline{\zeta}_{\eta,t<0}$ unchanged — which is something we can do just as easily whether η does or doesn't have any "internal parameters" in the first place.

So in quantifying the performance of η for behavior given by $\underline{\zeta}$ we compare $\underline{\zeta}$ to a set of $\underline{\zeta}'$, a set restricted to those $\underline{\zeta}'$ sharing $\underline{\zeta}$'s past: $\underline{\zeta}'_{,t<0} = \underline{\zeta}_{,t<0}$, $\underline{\zeta}'_{\eta,0} = \underline{\zeta}_{\eta,0}$, and $\underline{\zeta}'_{,t\geq 0} \in C_{,t\geq 0}$. Since $\underline{\zeta}'_{\eta,0}$ is free to vary (reflecting the possible changes in the state of η at time 0), $\underline{\zeta}' \notin C$, in general, and we may even wish to allow $\underline{\zeta}'_{,t\geq 0} \notin C_{,t\geq 0}$ in certain circumstances. (Recall that C may reflect other restrictions imposed on allowed worldlines besides adherence to the underlying dynamical laws, so simply obeying those laws does not force a worldline to lie on C.) However our presumption is that as far as utility values are concerned, considering such $\underline{\zeta}'$ is equivalent to considering a more restricted set of $\underline{\zeta}'$ with "modified internal parameters", all of which are $\in C$.

We now present a formalization of this performance measure. Given C and a measure $d\mu(\underline{\zeta}_{,0})$ demarcating what points in $\underline{\mathbf{Z}}_{\eta,0}$ we're interested in, we define the ($t = 0$) **intelligence** for node η of a point $\underline{\zeta}$ with respect to a utility U as follows:

$$\epsilon_{\eta,U}(\underline{\zeta}_{,}) \equiv \int d\mu(\underline{\zeta}'_{,0}) \;\; \Theta[U(\underline{\zeta}) - U(\underline{\zeta}_{,t<0}, C(\underline{\zeta}'_{,0}))] \times \delta(\underline{\zeta}'_{\eta,0} - \underline{\zeta}_{\eta,0}) \tag{1}$$

where $\Theta(.)$ is the Heaviside theta function which equals 0 if its argument is below 0 and equals 1 otherwise, $\delta(.)$ is the Dirac delta function, and we assume that $\int d\mu(\underline{\zeta}'_{\eta,0}) = 1$.

Intuitively, $\epsilon_{\eta,U}(\underline{\zeta})$ measures the fraction of alternative states of η which, if η had been in those states at time 0, would degrade η's performance (as measured by U). As an example, conventional full rationality game theory involving Nash equilibria is exclusively concerned with scenarios in which all such fractions equal 1.[8] More generally, competent greedy pursuit of private utility U by the microlearner controlling node η means that the intelligence of η for personal utility U, $\epsilon_{\eta,U}(\underline{\zeta})$, is close to 1. Accordingly, we will often refer interchangeably to a capable microlearner's "pursuing private utility U", and to its having high intelligence for personal utility U. Alternatively, if the microlearner for node η is incompetent, then it may even be that "by luck" its intelligence for some personal

[8] As an alternative to such fully rational games, one can define a bounded rational game as one in which the intelligences equal some vector $\vec{\epsilon}$ whose components need not all equal 1. Many of the theorems of conventional game theory can be directly carried over to apply to such bounded-rational games [220] by redefining the utility functions of the players. I.e., much of conventional full rationality game theory applies even to games with bounded rationality, under the appropriate transformation. This potentially has major implications for the common criticism of modern economic theory that its full rationality assumption does not hold in the real world.

utility $\{g_\eta\}$ exceeds its intelligence for the different private utility that it's actually trying to maximize, U_η.

Any two utility functions that are related by a monotonically increasing transformation reflect the same preference ordering over the possible arguments of those functions. Since it is only that ordering that we are are ever concerned with, we would like to remove this degeneracy by "normalizing" all utility functions. To see what this means in the COIN context, fix $\underline{\zeta}_{\hat{\eta},}$. Viewed as a function from $\underline{\mathbf{Z}}_{\eta,} \to \mathcal{R}$, $\epsilon_{\eta,U}(\underline{\zeta}_{\hat{\eta},}, .)$ is itself a utility function. It says how well η would have performed for all points $\underline{\zeta}_{\eta,}$. Accordingly, the integral transform taking U to $\epsilon_{\eta,U}(\underline{\zeta}_{\hat{\eta},}, .)$ is a (contractive, non-invertible) mapping from utilities to utilities. It can be proven that any mapping from utilities to utilities that meets certain simple desiderata must be such an integral transform. (An example of such a desideratum is that the mapping has the same output utility for any two input utilities that are monotonically increasing transforms of one another.) In this, intelligence is the unique way of "normalizing" Von Neumann utility functions.

4.2.3 Learnability

Intelligence can be a difficult quantity to work with, unfortunately. As an example, fix η, and consider any (small region centered about some) $\underline{\zeta}$ that is not a local maximum of some utility U. Then by increasing the values of U evaluated in that small region we will increase the intelligence $\epsilon_{\eta,U}(\underline{\zeta})$. However in doing this we will also necessarily *decrease* the intelligence at points outside that region. So intelligence has a non-local character, a character that prevents us from directly modifying it to ensure that it is simultaneously high for any and all $\underline{\zeta}$.

A second, more general problem is that without specifying the details of a microlearner, it can be extremely difficult to predict which of two private utilities the microlearner will be better able to learn. (Indeed, even *with* the details, making that prediction can be nearly impossible.) So it can be extremely difficult to determine what private utility intelligence values will accrue to various choices of those private utilities. In other words, macrolearning that involves modifying the private utilities to try to increase directly intelligence with respect to those utilities can be quite difficult.

Fortunately we can circumvent many of these difficulties by using a proxy for (private utility) intelligence. Although we expect its value usually to be correlated with that of intelligence in practice, this proxy does not share intelligence's non-local nature. In addition, the proxy does not depend heavily on the details of the microlearning algorithms used, i.e., it is fairly independent of those aspects of C.

We motivate this proxy by considering having $g_\eta = G$ for all η. If we try to actually use these $\{g_\eta\}$ as the microlearners' private utilities, particularly if the COIN is large, we will invariably encounter a very bad signal-to-noise problem. For this choice of utilities, the effects of the actions taken by node η on its utility may be "swamped" and effectively invisible, since there are so many other processes going into determining G's value. In such a scenario, there is nothing that η's microlearner can do to reliably achieve high intelligence.[9]

One natural way to quantify this effect is as **(utility) learnability**: Given a measure $d\mu(\underline{\zeta}'_{,0})$ and manifold C, the utility learnability of a utility U for a node η at $\underline{\zeta}$ is:

$$\Lambda_{\eta,U}(\underline{\zeta}) \equiv \frac{\int d\mu(\underline{\zeta}'_{,0}) \ \ |U(\underline{\zeta}_{,t<0}, C(\underline{\zeta}_{\hat{\eta},0}, \underline{\zeta}'_{\eta,0})) - U(\underline{\zeta})|}{\int d\mu(\underline{\zeta}'_{,0}) \ \ |U(\underline{\zeta}_{,t<0}, C(\underline{\zeta}'_{\hat{\eta},0}, \underline{\zeta}_{\eta,0})) - U(\underline{\zeta})|} \ . \tag{2}$$

(**Intelligence learnability** is defined the same way, with $U(.)$ replaced by $\epsilon_{\eta,U}(.)$.) Note that scaling all utility values by the same overall factor does not affect the value of the learnability.

The integrand in the numerator of the definition of learnability reflects how much of the change in U that results from replacing $\underline{\zeta}_{,0}$ with $\underline{\zeta}'_{,0}$ is due to the change in η's $t = 0$ state (the "signal"). The denominator reflects how much of the change in U that results from replacing $\underline{\zeta}$ with $\underline{\zeta}'$ is due to the change in the $t = 0$ states of nodes other than η (the "noise"). So learnability quantifies how easy it is for the microlearner to discern the "echo" of its behavior in the utility function U. Our presumption is that the microlearning algorithm will achieve higher intelligence if provided with a more learnable private utility.

Note that a particular value of utility learnability, by itself, has no significance. Simply rescaling the units of $\underline{\zeta}_{\eta,0}$ will change that value. Rather what is important is the ratio of differential learnabilities, at the same $\underline{\zeta}$, for different U's. Such a ratio quantifies the relative preferability of those U's.

More generally, learnability is not meant to capture all factors that will affect how high an intelligence value a particular microlearner will achieve. This is not possible if for no other reason then the fact that there are many such factors that are idiosyncratic to the microlearner used. In addition though, certain more general factors affecting learning, like the curse of dimensionality, are not explicitly designed into learnability. Learnability is not meant to quantify performance — that is what intelligence is designed to do. Rather (relative) learnability is meant to provide a guide for how to improve performance.

The (utility) **differential learnability** at a point $\underline{\zeta}$ is the learnability with $d\mu$ restricted to an infinitesimal ball about $\underline{\zeta}$. We formalize it as the following ratio of magnitudes of gradients:

$$\lambda_{\eta,U}(\underline{\zeta}) \equiv \frac{||\partial_{\underline{\zeta}_{\eta,0}} U(\underline{\zeta}_{,t<0}, C(\underline{\zeta}_{,0}))||}{||\partial_{\underline{\zeta}_{\hat{\eta},0}} U(\underline{\zeta}_{,t<0}, C(\underline{\zeta}_{,0}))||} \ . \tag{3}$$

One nice feature of differential learnability is that unlike learnability, it does not depend on choice of some measure $d\mu(.)$. This independence can lead to troubles if

[9] This "signal-to-noise" problem is actually endemic to reinforcement learning as a whole, even sometimes occurring when one has just a single reinforcement learner, and only a few random variables jointly determining the value of the rewards [54].

one is not careful however, and in particular if one uses learnability for purposes than choosing between utility functions. For example, in some situations, the COIN designer will have the option of enlarging the set of variables from the rest of the COIN that are "input" to some node η and that therefore can be used by η to decide what action to take. Intuitively, doing so will not affect the RL "signal" for η's microlearner (the magnitude of the potential "echo" of η's actions are not modified by changing some aspect of how it chooses among those actions). However it *will* reduce the "noise". In the full integral version of learnability, this effect can be captured by shrinking the support of $d\mu(.)$ to reflect the fact that the extra inputs to η at $t = 0$ are correlated with the $t = 0$ state of the external system. In differential learnability however this is not possible, precisely because no measure $d\mu(.)$ occurs in differential learnability. So we must capture the reduction in noise in some other fashion.[10]

A system that has infinite (differential, intelligence) learnability is said to be "perfectly" (differential, intelligence) learnable. It is straight-forward to prove that a system is perfectly learnable $\forall \zeta \in C$ iff $\forall \eta, g_\eta(\underline{\zeta})$ can be written as $\psi_\eta(\underline{\zeta}_{\eta,0})$ for some function $\psi_\eta(.)$. (See the discussion below on the general condition for a system's being perfectly factored.)

4.3 A descriptive framework for COINs

With these definitions in hand, we can now present (a portion of) one descriptive framework for COINs. In this subsection, after discussing salient characteristics in general, we present some theorems concerning the relationship between personal utilities and the salient characteristic we choose to concentrate on. We then discus how to use those theorems to induce that salient characteristic.

4.3.1 Candidate salient characteristics of a COIN

The starting point with a descriptive framework is the identification of "salient characteristics of a COIN which one strongly expects to be associated with its having large world utility". In this chapter we will focus on salient characteristics that concern the relationship between personal and world utilities. These characteristics are formalizations of the intuition that we want COINs in which the competent greedy pursuit of their private utilities by the microlearners results in large world utility, without any bottlenecks, TOC, "frustration" (in the spin glass sense) or the like.

One natural candidate for such a characteristic, related to Pareto optimality [?], is

[10] An example of how to do so is to replace $\partial_{\underline{\zeta}_{\eta,0}} U(\underline{\zeta}_{,t<0}, C(\underline{\zeta}_{,0}))$ in the definition of differential learnability with the projection of $\partial_{\underline{\zeta}_{\eta,0}} U(\underline{\zeta}_{,t<0}, C(\underline{\zeta}_{,0}))$ onto the tangent plane of $C_{,t\geq 0}$ at $\underline{\zeta}$. Assume that in addition to the restriction of obeying the dynamic laws $C(.)$ for evolution to times past $t = 0$, the manifold $C_{,t\geq 0}$ reflects the restriction on $\underline{\zeta}_{,t\geq 0}$ that the extra inputs to η at $t = 0$ are correlated with the $t = 0$ state of the external system. Under these circumsantaces, this projection of the gradient of the $\hat{\eta}$ components will reduce the noise term in the appropriate fashion. See [219].

weak triviality. It is defined by considering any two worldlines $\underline{\zeta}$ and $\underline{\zeta}'$ both of which are consistent with the system's dynamics (i.e., both of which lie on C), where for every node η, $g_\eta(\underline{\zeta}) \geq g_\eta(\underline{\zeta}')$. (An obvious variant is to restrict $\underline{\zeta}'_{,t<0} = \underline{\zeta}_{,t<0}$, and require only that both of the "partial vectors" $\underline{\zeta}'_{,t\geq 0}$ and $\underline{\zeta}_{,t\geq 0}$ obey the relevant dynamical laws, and therefore lie in $C_{,t\geq 0}$.) If for any such pair of worldlines it is necessarily true that $G(\underline{\zeta}) \geq G(\underline{\zeta}')$, we say that the system is weakly trivial. We might expect that systems that are weakly trivial for the microlearners' private utilities are configured correctly for inducing large world utility. After all, for such systems, if the microlearners collectively change $\underline{\zeta}$ in a way that ends up helping all of them, then necessarily the world utility also rises.

As it turns out though, weakly trivial systems can readily evolve to a world utility *minimum*, one that often involves TOC. To see this, consider automobile traffic in the absence of any traffic control system. Let each node be a different driver, and say their private utilities are how quickly they each individually get to their destination. Identify world utility as the sum of private utilities. Then by simple additivity, for all $\underline{\zeta}$ and $\underline{\zeta}'$, whether they lie on C or not, if $g_\eta(\underline{\zeta}) \geq g_\eta(\underline{\zeta}') \;\; \forall \eta$ it follows that $G(\underline{\zeta}) \geq G(\underline{\zeta}')$; the system is weakly trivial. However as any driver on a rush-hour freeway with no carpool lanes or metering lights can attest, every driver's pursuing their own goal definitely does not result in acceptable throughput for the system as a whole; modifications to private utility functions (like fines for violating carpool lanes or metering lights) would result in far better global behavior. A system's being weakly trivial provides no asurances regarding world utility.

The problem with weak triviality is precisely the fact that the individual microlearners *are* greedy. In a COIN, there is no system-wide incentive to replace $\underline{\zeta}$ with a different worldline that would improve everybody's private utility, as in the definition of weak triviality. Rather the incentives apply to each microlearner individually and motivate the learners to behave in a way that may well hurt some of them. So weak triviality is, upon examination, a poor choice for the salient characteristic of a COIN.

One alternative to weak triviality follows from considering that we must 'expect' a salient characteristic to be coupled to large world utility. of the definition of the descriptive framework. What can we reasonably assume about a running COIN? We cannot assume that all the private utilities will have large values — witness the traffic example. But we *can* assume that if the microlearners are well-designed, each of them will be doing close to as well it can *given the behavior of the other nodes*. In other words, within broad limits we can assume that the system is more likely to be in $\underline{\zeta}$ than $\underline{\zeta}'$ if for all η, $\epsilon_{\eta,g_\eta}(\underline{\zeta}) \geq \epsilon_{\eta,g_\eta}(\underline{\zeta}')$. We define a system to be **coordinated** iff for any such $\underline{\zeta}$ and $\underline{\zeta}'$ lying on C, $G(\underline{\zeta}) \geq G(\underline{\zeta}')$. (Again, an obvious variant is to restrict $\underline{\zeta}'_{,t<0} = \underline{\zeta}_{,t<0}$, and require only that both $\underline{\zeta}_{,t\geq 0}$ and $\underline{\zeta}'_{,t\geq 0}$ lie in $C_{,t\geq 0}$.) Traffic systems are *not* coordinated, in general. This is evident from the simple fact that if all drivers acted as though there were metering lights when in fact there weren't any, they would each be behaving with lower intelligence given the actions of the other drivers (each driver would benefit greatly by changing its behavior by no longer pretending there were metering lights, etc.). But nonetheless, world utility would be higher.

4.3.2 The Salient Characteristic of Factoredness

Like weak triviality, coordination is intimately related to the economics concept of Pareto optimality. Unfortunately, there is not room in this chapter to present the mathematics associated with coordination and its variants. However there is room to discuss a third candidate salient characteristic of COINs, one which like coordination (and unlike weak triviality) we can reasonably expect to be associated with large world utility. This alternative fixes weak triviality not by replacing the personal utilities $\{g_\eta\}$ with the intelligences $\{\epsilon_{\eta,g_\eta}\}$ as coordination does, but rather by only considering worldlines whose difference at time 0 involves a single node. This results in its being related to Nash equilibria rather than Pareto optimality.

Say that our COIN's worldline is $\underline{\zeta}$. Let $\underline{\zeta}'$ be any other worldline where $\underline{\zeta}_{,t<0} = \underline{\zeta}'_{,t<0}$, and where $\underline{\zeta}'_{,t\geq 0} \in C_{,t\geq 0}$. Now restrict attention to those $\underline{\zeta}'$ where at $t = 0$ $\underline{\zeta}$ and $\underline{\zeta}'$ differ only for node η. If for all such $\underline{\zeta}'$

$$sgn[g_\eta(\underline{\zeta}) - g_\eta(\underline{\zeta}_{,t<0}, C(\underline{\zeta}'_{,0}))] = sgn[G(\underline{\zeta}) - G(\underline{\zeta}_{,t<0}, C(\underline{\zeta}'_{,0}))] \ , \tag{4}$$

and if this is true for all nodes η, then we say that the COIN is **factored** for all those utilities $\{g_\eta\}$ (at $\underline{\zeta}$, with respect to time 0).

For a factored system, for any node η, *given the rest of the system*, if the node's state at $t = 0$ changes in a way that improves that node's utility over the rest of time, then it necessarily also improves world utility. Colloquially, for a system that is factored for a particular microlearner's private utility, if that learner does something that improves that personal utility, then everything else being equal, it has also done something that improves world utility. Of two potential microlearners for controlling node η (i.e., two potential $\underline{\zeta}_\eta$) whose behavior until $t = 0$ is identical but which disagree there, the microlearner that is smarter with respect to g will always result in a larger g, by definition of intelligence. Accordingly, for a factored system, the smarter microlearner is also the one that results in better G. So as long as we have deployed a sufficiently smart microlearner on η, we have assured a good G (given the rest of the system). Formally, this is expressed in the fact [219] that for a factored system, for all nodes η,

$$\epsilon_{\eta,g_\eta}(\underline{\zeta}) = \epsilon_{\eta,G}(\underline{\zeta}) \ . \tag{5}$$

One can also prove that Nash equilibria of a factored system are local maxima of world utility. Note that in keeping with our behaviorist perspective, nothing in the definition of factored requires private utilities. Indeed, it may well be that a system having private utilities $\{U_\eta\}$ is factored, but for personal utilities $\{g_\eta\}$ that differ from the $\{U_\eta\}$.

A system's being factored does *not* mean that a change to $\underline{\zeta}_{\eta,0}$ that improves $g_\eta(\underline{\zeta})$ cannot also hurt $g_{\eta'}(\underline{\zeta})$ for some $\eta' \neq \eta$. Intuitively, for a factored system, the side effects on the rest of the system of η's increasing its own utility do not end up decreasing world utility — but can have arbitrarily adverse effects on other private utilities. For factored systems, the separate microlearners successfully pursuing their separate goals do not frustrate each other *as far as world utility is concerned.*

In general, we can't have both perfect learnability and perfect factoredness. As an example, say that $\forall t, \underline{\mathbf{Z}}_{\eta,t} = \underline{\mathbf{Z}}_{\hat{\eta},t} = \mathcal{R}$. Then if $G(C(\underline{\zeta}_{,0})) = \underline{\zeta}_{\eta,0} \times \underline{\zeta}_{\hat{\eta},0}$ and the system is perfectly learnable, it is not perfectly factored. This is because $\underline{\zeta}_{\eta,0} = G(\underline{\zeta})/\underline{\zeta}_{\hat{\eta},0}$ for this case, and therefore perfect learnability requires that $\forall \underline{\zeta} \in C, g_\eta(\underline{\zeta}) = \psi_\eta(G(\underline{\zeta})/\underline{\zeta}_{\hat{\eta},0})$ for some function $\psi_\eta(.)$. However the partial derivative of this with respect to G will be negative for negative $\underline{\zeta}_{\hat{\eta},0}$, which means the system is actually "anti-factored" for such $\underline{\zeta}_{\hat{\eta},0}$. Due to such incompatibility between perfect factoredness and perfect learnability, we must usually be content with having high degree of factoredness and high learnability. In such situations, the emphasis of the macrolearning process should be more and more on having high degree of factoredness as we get closer and closer to a Nash equilibrium. This way the system won't relax to an incorrect local maximum.

In practice of course, a COIN will often not be perfectly factored. Nor in practice are we always interested only in whether the system is factored at one particular point (rather than across a region say). These issues are discussed in [219], where in particular a formal definition of of the **degree of factoredness** of a system is presented.

If a system is factored for utilities $\{g_\eta\}$, then it is also factored for any utilities $\{g'_\eta\}$ where for each η g'_η is a monotonically increasing function of g_η. More generally, the following result characterizes the set of all factored personal utilities:

Theorem 1: A system is factored at all $\underline{\zeta} \in C$ iff for all those $\underline{\zeta}$, $\forall \eta$, we can write

$$g_\eta(\underline{\zeta}) = \Phi_\eta(\underline{\zeta}_{,t<0}, \underline{\zeta}_{\hat{\eta},0}, G(\underline{\zeta})) \tag{6}$$

for some function $\Phi_\eta(.,.,.)$ such that $\partial_G \Phi_\eta(\underline{\zeta}, \underline{\zeta}_{,t<0}, G) > 0$ for all $\underline{\zeta} \in C$ and associated G values. (The form of the $\{g_\eta\}$ off of C is arbitrary.)

Proof: For fixed $\underline{\zeta}_{\eta,0}$ and $\underline{\zeta}_{,t<0}$, any change to $\underline{\zeta}_{\eta,0}$ which keeps $\underline{\zeta}_{,t\geq 0}$ on C and which at the same time increases $G(\underline{\zeta}) = G(\underline{\zeta}_{,t<0}, C(\underline{\zeta}_{\hat{\eta},0}, \underline{\zeta}_{\eta,0}))$ must increase $\Phi_\eta(\underline{\zeta}_{,t<0}, \underline{\zeta}_{\hat{\eta},0}, G(\underline{\zeta}))$, due to the restriction on $\partial_G \Phi_\eta(\underline{\zeta}_{,t<0}, \underline{\zeta}_{\hat{\eta},0}, G)$. This establishes the backwards direction of the proof.

For the forward direction, write $g_\eta(\underline{\zeta}) = g_\eta(\underline{\zeta}, G(\underline{\zeta})) = g_\eta(\underline{\zeta}_{,t<0}, C(\underline{\zeta}_{\hat{\eta},0}, \underline{\zeta}_{\eta,0}), G(\underline{\zeta}))\ \ \forall\, \underline{\zeta} \in C$. Define this formulation of g_η as $\Phi_\eta(\underline{\zeta}_{,t<0}, \underline{\zeta}_{,0}, G(\underline{\zeta}))$, which we can re-express as $\Phi_\eta(\underline{\zeta}_{,t<0}, \underline{\zeta}_{\hat{\eta},0}, \underline{\zeta}_{\eta,0}, G(\underline{\zeta}))$. Now since the system is factored, $\forall \underline{\zeta} \in C, \forall \underline{\zeta}'_{,t\geq 0} \in C_{,t\geq 0}$,

$$\Phi_\eta(\underline{\zeta}_{,t<0}, \underline{\zeta}_{\hat{\eta},0}, \underline{\zeta}_{\eta,0}, G(C(\underline{\zeta}_{\hat{\eta},0}, \underline{\zeta}_{\eta,0}))) \;=\; \Phi_\eta(\underline{\zeta}_{,t<0}, \underline{\zeta}_{\hat{\eta},0}, \underline{\zeta}'_{\eta,0}, G(C(\underline{\zeta}_{\hat{\eta},0}, \underline{\zeta}'_{\eta,0})))$$

$$\Longleftrightarrow$$

$$G(\underline{\zeta}_{,t<0}, C(\underline{\zeta}_{\hat{\eta},0}, \underline{\zeta}_{\eta,0}))) \;=\; G(\underline{\zeta}_{,t<0}, C(\underline{\zeta}_{\hat{\eta},0}, \underline{\zeta}'_{\eta,0}))\ .$$

So consider any situation where the system is factored, and both the values of G and of $\underline{\zeta}_{\eta,0}$ are specified. Then we can find *any* $\underline{\zeta}_{\eta,0}$ consistent with those values (i.e., such that our provided value of G equals $G(\underline{\zeta}_{,t<0}, C(\underline{\zeta}_{\eta,0}, \underline{\zeta}_{\eta,0})))$, evaluate the resulting value of $\Phi_\eta(\underline{\zeta}_{,t<0}, \underline{\zeta}_{\eta,0}, \underline{\zeta}_{\eta,0}, G)$, and know that we would have gotten the same value if we had found a different consistent $\underline{\zeta}_{\eta,0}$. This is true for all $\underline{\zeta} \in C$. Therefore the mapping $(\underline{\zeta}_{,t<0}, \underline{\zeta}_{\eta,0}, G) \rightarrow \Phi_\eta$ is single-valued, and we can write $\Phi_\eta(\underline{\zeta}_{,t<0}, \underline{\zeta}_{\hat{\eta},0}, G(\underline{\zeta}))$. **QED.**

By Thm. 1, we can ensure that the system is factored without any concern for C, by having each $g_\eta(\underline{\zeta}) = \Phi_\eta(\underline{\zeta}_{,t<0}, \underline{\zeta}_{\hat{,}0}, G(\underline{\zeta})) \quad \forall \underline{\zeta} \in \underline{\mathbf{Z}}$. Alternatively, by only requiring that $\forall \underline{\zeta} \in C$ does $g_\eta(\underline{\zeta}) = \Phi_\eta(\underline{\zeta}_{,t<0}, \underline{\zeta}_{\hat{\eta},0}, G(\underline{\zeta}))$ (i.e., does $g_\eta(\underline{\zeta}_{,t<0}, C(\underline{\zeta}_{,0})) = \Phi_\eta(\underline{\zeta}_{,t<0}, \underline{\zeta}_{\eta,0}, G(\underline{\zeta}_{,t<0}, C(\underline{\zeta}_{,0}))))$, we can access a broader class of factored utilities, a class that *does* depend on C. Loosely speaking, for those utilities, we only need the projection of $\partial_{\underline{\zeta}_{,t\geq 0}} G(\underline{\zeta})$ onto $C_{\eta,0}$ to be parallel to the projection of $\partial_{\underline{\zeta}_{,t\geq 0}} g_\eta(\underline{\zeta})$ onto $C_{\eta,0}$. Given G and C, there are infinitely many $\partial_{\underline{\zeta}_{,t\geq 0}} g_\eta(\underline{\zeta})$ having this projection (the set of such $\partial_{\underline{\zeta}_{,t\geq 0}} g_\eta(\underline{\zeta})$ form a linear subspace of $\underline{\mathbf{Z}}$). The partial differential equations expressing the precise relationship are discussed in [219].

As an example of the foregoing, consider a "team game" (also known as an "exact potential game" [69, 156]) in which $g_\eta = G$ for all η. Such COINs are factored, trivially, regardless of C; if g_η rises, then G must as well, by definition. (Alternatively, to confirm that team games are factored just take $\Phi_\eta(\underline{\zeta}_{,t<0}, \underline{\zeta}_{\eta,0}, G) = G \quad \forall \eta$ in Thm. 1.) On the other hand, as discussed below, COINs with 'wonderful life' personal utilities are also factored, but the definition of such utilities depends on C.

4.3.3 Wonderful life utility

Due to their often having poor learnability and requiring centralized communication (among other infelicities), in practice team game utilities often are poor choices for personal utilities. Accordingly, it is often preferable to use some other set of factored utilities. To present an important example, first define the $(t = 0)$ **effect set** of node η at $\underline{\zeta}$, $C^{eff}_\eta(\underline{\zeta})$, as the set of all components $\underline{\zeta}_{\eta',t}$ for which $\partial_{\underline{\zeta}_{\eta,0}}(C(\underline{\zeta}_{,0}))_{\eta',t} \neq \vec{0}$. Define the effect set C^{eff}_η with no specification of $\underline{\zeta}$ as $\cup_{\underline{\zeta} \in C} C^{eff}_\eta(\underline{\zeta})$. (We take this latter definition to be the default meaning of "effect set".) Intuitively, η's effect set is the set of all components $\underline{\zeta}_{\eta',t}$ which would be affected by a change in the state of node η at time 0. (They may or may not be affected by changes in the $t = 0$ states of the other nodes.) The extension for times other than 0 is immedaite, but will be skipped here to minimize the number of variables we must keep track of.

Next take the **Wonderful Life** set σ to be a set of components (η', t), and define $\mathrm{CL}_\sigma(\underline{\zeta})$ as the vector $\underline{\zeta}$ modified by clamping the σ-components of $\underline{\zeta}$ to an arbitrary fixed value, here taken to be $\vec{0}$. Then the the value of the **wonderful life utility** (WLU for short) for σ at $\underline{\zeta}$ is:

$$WLU_\sigma(\underline{\zeta}) \equiv G(\underline{\zeta}) - G(\mathrm{CL}_\sigma(\underline{\zeta})) \ . \tag{7}$$

In particular, the WLU for the effect set of node η is $G(\underline{\zeta}) - G(\mathrm{CL}_{C^{eff}_\eta}(\underline{\zeta}))$, which for $\underline{\zeta} \in C$ can be written as $G(\underline{\zeta}_{,t<0}, C(\underline{\zeta}_{,0})) - G(\mathrm{CL}_{C^{eff}_\eta}(\underline{\zeta}_{,t<0}, C(\underline{\zeta}_{,0})))$.

We can view η's effect set WLU as analogous to the change in world utility that would have arisen if node η "had never existed". (Hence the name of this utility - cf. the Frank Capra movie.) Note however, that CL is a purely mathematical operation. Indeed, no assumption is even being made that $\mathrm{CL}_\sigma(\underline{\zeta})$ is consistent with the dynamics of the system. The sequence of states the node η is clamped to in the definition of the WLU need not be consistent with the dynamical laws of the system.

This dynamics-independence is a crucial strength of the WLU. It means that to evaluate the WLU we do *not* try to infer how the system would have evolved if node η's state were set to 0 at time 0 and the system evolved from there. So long as we know $\underline{\zeta}$ extending over all time, and so long as we know G, we know the value of WLU. This is true even if we know nothing of the dynamics of the system.

An important example is effect set wonderful life utilities when the set of all nodes is partitioned into 'subworld' in such a way that all nodes in the same subworld ω share substantially the same effect set. In such a situation, all nodes in the same subworld ω will have essentially the same personal utilities, exactly as they would if they used team game utilities with a "world" given by ω. When all such nodes have large intelligence values, this sharing of the personal utility will mean that all nodes in the same subworld are acting in a coordinated fashion, loosely speaking.

The importance of the WLU arises from the following results:

Theorem 2: A COIN is factored for personal utilities set equal to the associated effect set wonderful life utilities.

Proof: Since $\mathrm{CL}_{C^{eff}_\eta}(\underline{\zeta})$ is independent of $\underline{\zeta}_{\eta',t}$ for all $(\eta', t) \in C^{eff}_\eta$, so is the $\underline{\mathbf{Z}}$ vector $\mathrm{CL}_{C^{eff}_\eta}(\underline{\zeta}_{,t<0}, C(\underline{\zeta}_{,0}))$. I.e., $\partial_{\underline{\zeta}_{\eta,0}}[\mathrm{CL}_{C^{eff}_\eta}(\underline{\zeta}_{,t<0}, C(\underline{\zeta}_{,0}))]_{\eta',t} = \vec{0} \;\; \forall \eta', t$. This means that viewed as a function from $C_{,0}$ to $\underline{\mathbf{Z}}$, $\mathrm{CL}_{C^{eff}_\eta}(\underline{\zeta}_{,t<0}, C(.))$ is a single-valued function of $\underline{\zeta}_{,\eta,0}$. Therefore $G(\mathrm{CL}_{C^{eff}_\eta}(\underline{\zeta}_{,t<0}, C(\underline{\zeta}_{,0})))$ can only depend on $\underline{\zeta}_{,t<0}$ and the non-η components of $\underline{\zeta}_{,0}$. Accordingly, the WLU for C^{eff}_η is just G minus a term that is a function of $\underline{\zeta}_{,t<0}$ and $\underline{\zeta}_{,\eta,0}$. By choosing $\Phi_\eta(.,.,.)$ in Thm. 1 to be that difference, we see that η's effect set WLU is of the form necessary for the system to be factored. **QED.**

More generally, the system is factored if each node η's personal utility is (a monotonically increasing function of) the WLU for a set σ_η that contains C^{eff}_η.

To understand the potential practical advantages of the WLU, we start with the following:

Theorem 3:

$$\frac{\lambda_{\eta,WLU_{C^{eff}_\eta}}(\underline{\zeta})}{\lambda_{\eta,G}(\underline{\zeta})} = \frac{||\partial_{\underline{\zeta}_{\hat{\eta},0}} G(C(\underline{\zeta}_{,0}))||}{||\partial_{\underline{\zeta}_{\hat{\eta},0}} G(C(\underline{\zeta}_{,0})) - \partial_{\underline{\zeta}_{\hat{\eta},0}} G(\mathrm{CL}^{eff}_\eta(C(\underline{\zeta}_{,0})))||} \; .$$

Proof: Writing it out,

$$\lambda_{\eta,WLU_{C^{eff}_\eta}}(\underline{\zeta}) = \frac{\partial_{\underline{\zeta}_{\eta,0}} G(C(\underline{\zeta}_{,0})) - \partial_{\underline{\zeta}_{\eta,0}} G(\mathrm{CL}^{eff}_\eta(C(\underline{\zeta}_{,0})))}{\partial_{\underline{\zeta}_{\hat{\eta},0}} G(C(\underline{\zeta}_{,0})) - \partial_{\underline{\zeta}_{\hat{\eta},0}} G(\mathrm{CL}^{eff}_\eta(C(\underline{\zeta}_{,0})))} \; .$$

The second term in the numerator equals 0, by definition of effect set. Dividing by the similar expression for $\lambda_{\eta,G}(\underline{\zeta})$ then gives the result claimed. **QED.**

So if we expect that ratio of magnitudes of gradients to be large, effect set WLU has much higher learnability than team game utility — while still being factored, like team game utility. As an example, consider the case where the COIN is a very large system, with η being only a relatively minor part of the system (e.g., a large human economy with η being a "typical John Doe"). Often in such a system, for the vast majority of nodes $\eta' \neq \eta$, how G varies with $\underline{\zeta}_{\eta'}$, will be essentially independent of the value $\underline{\zeta}_{\eta,0}$. (E.g., how GDP of the US economy varies with the actions of John Doe in Peoria, Illinois will be independent of the state of some Jane Smith in Los Angeles, California.) In such circumstances, Thm. 3 tells us that the effect set utility for η will have a far larger learnability than does the world utility.

Next consider the case where, for some node η, we can write $G(\underline{\zeta})$ as $G_1(\underline{\zeta}_{C^{eff}_\eta}) + G_2(\underline{\zeta}_{,t<0}, \underline{\zeta}_{\hat{C}^{eff}_\eta})$. Say it is also true that η's effect set is a small fraction of the set of all components. In this case it often true that the values of $G(.)$ are much larger than those of $G_1(.)$, which means that partial derivatives of $G(.)$ are much larger than those of $G_1(.)$. In such situations the effect set WLU is far more learnable than the world utility, due to the following results:

Theorem 4: If for some node η there is a set σ containing C^{eff}_η, a function $G_1(\underline{\zeta}_\sigma \in \underline{\mathbf{Z}}_\sigma)$, and a function $G_2(\underline{\zeta}_{\hat{\sigma}} \in \underline{\mathbf{Z}}_{\hat{\sigma}})$, such that $G(\underline{\zeta}) = G_1(\underline{\zeta}_\sigma) + G_2(\underline{\zeta}_{\hat{\sigma}})$, then

$$\frac{\lambda_{\eta,WLU_\sigma}(\underline{\zeta})}{\lambda_{\eta,G}(\underline{\zeta})} = \frac{||\partial_{\underline{\zeta}_{\hat{\eta},0}} G(C(\underline{\zeta}_{,0}))||}{||\partial_{\underline{\zeta}_{\hat{\eta},0}} G(\mathrm{CL}_{\hat{\sigma}}(C(\underline{\zeta}_{,0})))||} \; .$$

Proof: For brevity, write G_1 and G_2 both as functions of full $\underline{\zeta} \in \underline{\mathbf{Z}}$, just such functions that are only allowed to depend on the components of $\underline{\zeta}$ that lie in σ and those components that do not lie in σ, respectively. Then the σ WLU for node η is just $g_\eta(\underline{\zeta}) = G_1(\underline{\zeta}) - G_1(\mathrm{CL}_\sigma(\underline{\zeta}))$. Since in that second term we're clamping all the components of $\underline{\zeta}$ that $G_1(.)$ cares about, for this personal utility $\partial_{\underline{\zeta}_{,0}} g_\eta(C(\underline{\zeta}_{,0})) = \partial_{\underline{\zeta}_{,0}} G_1(C(\underline{\zeta}_{,0}))$.

So in particular $\partial_{\underline{\zeta}_{\hat{\eta},0}} g_\eta(C(\underline{\zeta}_{,0})) = \partial_{\underline{\zeta}_{\hat{\eta},0}} G_1(C(\underline{\zeta}_{,0})) = \partial_{\underline{\zeta}_{\hat{\eta},0}} G(\mathrm{CL}_{\hat{\sigma}}(C(\underline{\zeta}_{,0})))$. Now by definition of effect set, $\partial_{\underline{\zeta}_{\eta,0}} G_2(\underline{\zeta}_{,t<0}, C(\underline{\zeta}_{,0})) = \vec{0}$, since $\hat{\sigma}$ does not contain C^{eff}_η. So $\partial_{\underline{\zeta}_{\eta,0}} G(C(\underline{\zeta}_{,0})) = \partial_{\underline{\zeta}_{\eta,0}} G_1(C(\underline{\zeta}_{,0})) = \partial_{\underline{\zeta}_{\eta,0}} g_\eta(C(\underline{\zeta}_{,0}))$. **QED.**

The obvious extensions of Thm.'s 3 and 4 for when we're considering effect sets with respect to times other than 0 holds.

By Thm. 4, when the conditions in that theorem hold the choice of clamping operation (i.e., the choice of the "arbitrary fixed value" the node is clamped to) is irrelevant as far as learnability is concerned. For other G's though the precise definition of CL(.) can affect the learnability of the effect set WLU.

An important special case of Thm. 4 is the following:

Corollary 1: If for some node η we can write:

i) $G(\underline{\zeta}) = G_1(\underline{\zeta}_{C^{eff}_\eta}) + G_2([\underline{\zeta}_{\hat{C}^{eff}_\eta}]_{t \geq 0}) + G_3(\underline{\zeta}_{,t<0})$,

and if

ii) $||\partial_{\underline{\zeta}_{\hat{\eta},0}} G(C(\underline{\zeta}_{,0}))|| \gg ||\partial_{\underline{\zeta}_{\hat{\eta},0}} G_1([C(\underline{\zeta}_{,0})]_{C^{eff}_\eta})||$,

then

$\lambda_{\eta, WLU_{C^{eff}_\eta}}(\underline{\zeta}) \gg \lambda_{\eta,G}(\underline{\zeta})$.

4.3.4 Inducing our salient characteristic

Usually in a descriptive framework our mathematics — a formal investigation of the salient characteristics — will not provide theorems of the sort, "If you modify the COIN the following way at time t, the the value of the world utility will increase." Rather it provides theorems that relate a COIN's salient characteristics with the general properties of the COIN's entire history, and in particular with those properties embodied in C. In particular, the salient characteristic that we are concerned with in this chapter is that the system be highly intelligent for personal utilities for which it is factored, and our mathematics concerns the relationship between factoredness, intelligence, personal utilities, effect sets, and the like.

More formally, the desideratum associated with our salient characteristic is that we want the COIN to be at a $\underline{\zeta}$ for which there is some set of $\{g_\eta\}$ (not necessarily consisting of private utilities) such that (a) $\underline{\zeta}$ is factored for the $\{g_\eta\}$, and (b) $\epsilon_{\eta,g_\eta}(\underline{\zeta})$ is large for all η. Now there are several ways one might try to induce the COIN to be at such a point. One approach is to have each algorithm controlling η explicitly try to "steer" the worldline towards such a point. In this approach η needn't even have a private utility in the usual sense. (The overt "goal" of the algorithm controlling η involves finding a $\underline{\zeta}$ with a good associated extremum over the class of all possible g_η, independent of any private utilities.) Now initialization of the COIN, i.e., fixing of $\underline{\zeta}$, involves setting the algorithm controlling η, in this case to the steering algorithm. Accordingly, in this approach in initialization we fix $\underline{\zeta}$ to a point for which there is some special g_η such that both $\underline{\zeta}$ is factored for g_η, and $\epsilon_{\eta,g_\eta}(\underline{\zeta})$ is large. There is nothing peculiar about this. What is odd though is that in this approach we do not know what that "special" g_η is when we do that initialization; it's to be determined, by the unfolding of the system.

In this chapter we concentrate on a different approach, which can involve either initialization or macrolearning. In this alternative we deploy the $\{g_\eta\}$ as the microlearners' private utilities at some $t < 0$, in a process not captured in C, so as to induce a factored COIN that is as intelligent as possible. (It is with that "deploying of the $\{g_\eta\}$" that we are trying to induce our salient characteristic in the COIN.) Since in this approach we are using private utilities, we can replace intelligence with its surrogate, learnability. So our task is to choose $\{g_\eta\}$ which are as learnable as possible while still being factored.

Solving for such utilities can be expressed as solving a set of coupled partial differential equations. Those equations involve the tangent plane to the manifold C, a functional trading off (the differential versions of) degree of factoredness and learnability, and any communication constraints on the nodes we must respect. While there is not space n the current chapter to present those equations, we can note that they are highly dependent on the correlations among the components of $\underline{\zeta}_{\eta,t}$. So in this approach, in COIN initialization we use some preliminary guesses as to those correlations to set the initial $\{g_\eta\}$. For example, the effect set of a node constitutes all components $\underline{\zeta}_{\eta',t>0}$ that have non-zero correlation with $\underline{\zeta}_{\eta,0}$. Furthermore, by Thm. 2 the system is factored for effect set WLU personal utilities. And by Coroll. 1, for small effect sets, the effect set WLU has much greater differential utility learnability than does G. Extending the reasoning behind this result to all $\underline{\zeta}$ (or at least all likely $\underline{\zeta}$), we see that for this scenario, the descriptive framework advises us to use Wonderful Life private utilities based on (guesses for) the associated effect sets rather than the team game private utilities, $g_\eta = G \ \forall \eta$.

In macrolearning we must instead run-time estimate an approximate solution to our partial differential equations, based on statistical inference.[11] As an example, we might start with an initial guess as to η's effect set, and set its private utility to the associated WLU. But then as we watch the system run and observe the correlations among the components of $\underline{\zeta}$, we might modify which components we think comprise η's effect set, and modify η's personal utility accordingly.

[11] In the physical world, it is often useful to employ devices using algorithms that are based on probabilistic concepts, even when the underlying system is deterministic. (Indeed, theological Bayesians invoke a "degree of belief" interpretation of probability to *demand* such an approach — see [?] for a dicussion of the legitimacy of this viewpoint.) Similarly, although we take the underlying system in a COIN to be deterministic, it is often useful to use microlearners or — as here — macrolearners that are based on probabilistic concepts.

4.4 Illustrative Simulations of our Descriptive Framework

As implied above, often one can perform reasonable COIN intialization and/or macrolearning without writing down the partial differential equations governing our salient characteristic explicitly. Simply "hacking" one's way to the goal of maximizing both degree of factoredness and intelligibility, for example by estimating effect sets, often results in dramatic improvement in performance. This is illustrated in the experiments recounted in the next two subsections.

4.4.1 COIN Initialization

Even if we don't exactly know the effect set of each node η, often we will be able to make a reasonable guess about which components of $\underline{\zeta}$ comprise the "preponderance" of η's effect set. We call such a set a **guessed effect set**. As an example, often the primary effects of changes to η's state will be on the future state of η, with only relatively minor effects on the future states of other nodes. In such situations, we would expect to still get good results if we approximated the effect set WLU of each node η with a WLU based on the guessed effect set $\underline{\zeta}_{\eta,t\geq 0}$. In other words, we would expect to be able to replace $\mathrm{WLU}_{C^{eff}_\eta}$ with $\mathrm{WLU}_{\underline{\zeta}_{\eta,t\geq 0}}$ and still get good performace.

This phenomenon was borne out in the experiments recounted in [221] that used COIN initialization for distributed control of network packet routing. In a conventional approach to packet routing, each router runs what it believes (based on the information available to it) to be a shortest path algorithm (SPA), i.e., each router sends its packets in the way that it surmises will get those packets to their destinations most quickly. Unlike with a COIN, with SPA-based routing the routers have no concern for the possible deleterious side-effects of their routing decisions on the global performance (e.g., they have no concern for whether they induce bottlenecks). We ran simulations in which we compared a COIN-based routing system to an SPA-based system. For the COIN-based system G was global throughput and no macrolearning was used. The COIN initialization was to have each router's private utility be a WLU based on an associated guessed effect set generated *a priori*. In addition, the COIN-based system was realistic in that each router's reinforcement algorithm had imperfect knowledge of the state of the system. On the other hand, the SPA was an idealized "best-possible" system, in which each router knew exactly what the shortest paths were at any given time. Despite the handicap that this disparity imposed on the COIN, it achieved significantly better global throughput in our experiments than did the perfect-knowledge SPA-based system.

The experiments in [221] were primarilly concerned with the application of packet-routing. To concentrate more precisely on the issue of COIN initializatin, we ran subsequent experiments on variants of Arthur's famous "El Farol bar problem" (see Section 3). To faciliate the analysis we modified Arthur's original problem to be more general, and since we were not interested in directly comparing our results to those in the literature, we used a more conventional (and arguably "dumber") machine learning algorithm than the ones investigated in [4, 40, 44, 57].

In this formulation of the bar problem, there are N agents, each of whom picks one of seven nights to attend a bar the following week, a process that is then repeated. In each week, each agent's pick is determined by its predictions of the associated rewards it would receive. These predictions in turn are based solely upon the rewards received by the agent in preceding weeks. An agent's "pick" at week t (i.e., its node's state at that week) is represented as a unary seven-dimensional vector. So η's zeroing its state in some week, as in the $\mathrm{CL}_{\underline{\zeta}_{\eta,t}}$ operation, essentially means it elects not to attend any night that week.

The world utility is

$$G(\underline{\zeta}) \;=\; \sum_t R(\underline{\zeta}_{,t}),$$

where: $R(\underline{\zeta}_{,t}) = \sum_{k=1}^{7} \gamma_k(x_k(\underline{\zeta}_{,t}))$; $x_k(\underline{\zeta}_{,t})$ is the total attendance on night k at week t; $\gamma_k(y) \equiv \alpha_k y \exp(-y/c)$; and c and each of the $\{\alpha_k\}$ are real-valued parameters. Intuitively, the "world reward" R is the sum of the global "rewards" for each night in each week. It reflects the effects in the bar as the attendance profile of agents changes. When there are too few agents attending some night, the bar suffers from lack of activity and therefore the global reward for that night is low. Conversely, when there are too many agents the bar is overcrowded and the reward for that night is again low. Note that $\gamma_k(\cdot)$ reaches its maximum when its argument equals c.

In these experiments we investigate two different $\vec{\alpha}$'s. One treats all nights equally; $\vec{\alpha} = [1\ 1\ 1\ 1\ 1\ 1\ 1]$. The other is only concerned with one night; $\vec{\alpha} = [0\ 0\ 0\ 7\ 0\ 0\ 0]$. In our experiments, $c = 6$ and N is chosen to be 4 times larger than the number of agents necessary to have c agents attend the bar on each of the seven nights, i.e., there are $4 \times 6 \times 7 = 168$ agents (this ensures that there are no trivial solutions and that for the world utility to be maximized, the agents have to "cooperate").

As explicated below, our microlearning algorithms worked by providing a real-valued "reward" signal to each agent at each week t. Each agent's reward function is a surrogate for an associated utility function for that agent. The difference between the two functions is that the reward function only reflects the state of the system at one moment in time (and therefore is potentially observable), whereas the utility function reflects the agent's ultimate goal, and therefore can depend on the full history of that agent across time.

We investigated three agent reward functions. With d_η the night selected by η, they are:

$$\begin{array}{lll}
\text{Uniform Division (UD):} & r_\eta(\underline{\zeta}_{,t}) & \equiv \gamma_{d_\eta}(x_{d_\eta}(\underline{\zeta}_{,t}))/x_{d_\eta}(\underline{\zeta}_{,t}) \\
\text{Global (G):} & r_\eta(\underline{\zeta}_{,t}) & \equiv R(\underline{\zeta}_{,t}) = \sum_{k=1}^{7} \gamma_k(x_k(\underline{\zeta}_{,t})) \\
\text{Wonderful Life (WL):} & r_\eta(\underline{\zeta}_{,t}) & \equiv R(\underline{\zeta}_{,t}) - R(\mathrm{CL}_{\underline{\zeta}_{\eta,t}}(\underline{\zeta}_{,t})) \\
 & & = \gamma_{d_\eta}(x_{d_\eta}(\underline{\zeta}_{,t})) - \gamma_{d_\eta}(x_{d_\eta}(\mathrm{CL}_{\underline{\zeta}_{\eta,t}}(\underline{\zeta}_{,t})))
\end{array}$$

The conventional UD reward is a natural "naive" choice for the agents' reward; the total reward on each night gets uniformly divided among the agents attending that night. If we take $g_\eta(\underline{\zeta}) = \sum_t r_\eta(\underline{\zeta}_{,t})$ (i.e., η's utility is an undiscounted sum of its rewards), then for the UD reward $G(\underline{\zeta}) = \sum_\eta \; g_\eta(\underline{\zeta})$, so that the system is weakly trivial. The original version of the bar problem in the physics literature [44] is the special case where UD reward is used but there are only two "nights" in the week (one of which corresponds to "staying at home"); $\vec{\alpha}$ is uniform; and $\gamma_k(x_k) = \Theta(N/2 - x_k)$. So the reward to agent η is 1 if it attends the bar and it is less than half full, or if it stays at home and the bar is more than half-full. Reward is 0 otherwise. (In addition, unlike in our COIN-based systems, in the original work on the bar problem the microlearners work by explicitly predicting whether the bar will be more than half full, rather than by directly modifying behavior to try to increase a reward signal.)

In contrast to the UD reward, providing the G reward at time t to each agent results in all agents receiving the same reward. For this reward function, the system is automatically factored if we define $g_\eta(\underline{\zeta}) \equiv \sum_t r_\eta(\underline{\zeta}_{,t})$. However, evaluation of this reward function requires centralized communication concerning all seven nights. Furthermore, given that there are 168 agents, G is likely to have poor learnability as a reward for any individual agent.

This latter problem is obviated by using the WL reward, where the subtraction of the clamped term removes some of the "noise" of the activity of all other agents, leaving only the underlying "signal" of how the agent in question affects the utility. So one would expect that with the WL reward the agents can readily discern the effects of their actions on their rewards. Even though the conditions in Coroll. 1 don't hold[12], this reasoning accords with the implicit advice of Coroll. 1 under the approximation of the $t = 0$ effect set as $C^{eff}_\eta \approx \underline{\zeta}_{\eta, t \geq 0}$. I.e., it agrees with that corollary's implicit advice under the identification of $\underline{\zeta}_{\eta, t \geq 0}$ as η's $t = 0$ guessed effect set.

In fact, we can readily calculate the ratio of the WL reward's learnability to that of the G reward, by recasting the system as existing for only a single instant so that $C^{eff}_\eta = \underline{\zeta}_{\eta,0}$ exactly and then applying Thm. 3. So for example, say that all $\alpha_k = 1$, and that the number of nodes N is evenly divided among the seven nights, then the numerator term in Thm. 3 is $|e^{(-N/7c)} \; [1 - N/7c] \; \sqrt{N-1}|$. The denomintor term in Thm. 3 under these conditions is $|e^{(-N/7c)} \; [1 - N/7c] \; [1 - e^{1/c}(1 - \frac{7}{N-7c})] \; \sqrt{\frac{N}{7} - 1}$. So the ratio is $|\sqrt{7 \frac{N-1}{N-7}} \; \frac{N-7c}{(N-7c)(1-e^{1/c}) + 7e^{1/c}}| = 38$.[13]

---

[12]The $t = 0$ elements of C^{eff}_η are just $\underline{\zeta}_{\eta,t=0}$, but the contributions of $\underline{\zeta}_{,t=0}$ to G cannot be written as a sum of a $\underline{\zeta}_{\eta,t=0}$ contribution and a $\underline{\zeta}_{\hat{\eta},t=0}$ contribution.

[13]Since λ is invaraint under rescaling, we can scale the UD reward by N and take σ to be all nodes attending the night agent η attends in the week at hand, so that $G(\underline{\zeta}) = G_1(\underline{\zeta}_\sigma) + G_2(\underline{\zeta}_{\hat{\sigma}}) = UD(\underline{\zeta}_\sigma) + G_2(\underline{\zeta}_{\hat{\sigma}})$. We can then apply Thm. 4, to see that $\lambda_{\eta,UD}(\underline{\zeta}) / \lambda_{\eta,UG}(\underline{\zeta}) = \sqrt{7 \frac{N-1}{N-7}}$ — which is about 14 times worse than $\lambda_{\eta,WLU}(\underline{\zeta}) / \lambda_{\eta,UG}(\underline{\zeta})$. This is in addition to the disadvantage of the UD reward that

In addition to this learnability advantage of the WL reward, to evaluate its WL reward each agent only needs to know the total attendance on the night it attended, so no centralized communication is required. Finally, although the system won't be perfectly factored for this reward (since in fact the effect set of η's action at t would be expected to extend a bit beyond $\underline{\zeta}_{\eta,t}$), one might expect that it is close enough to being factored to result in large world utility.

Each agent keeps a seven dimensional Euclidean vector representing its estimate of the reward for attending each night of the week. At the end of each week, the component of this vector corresponding to the night just attended is proportionally adjusted towards the actual reward just received. At the beginning of the succeeding week, the agent picks the night to attend using a Boltzmann distribution with energies given by the components of the vector of estimated rewards, where the temperature in the Boltzmann distribution decays in time. (This learning algorithm is equivalent to Claus and Boutilier's [51] independent learner algorithm for multi-agent reinforcement learning.) We used the same parameters (learning rate, Boltzmann temperature, decay rates, etc.) for all three reward functions. (This is an *extremely* primitive RL algorithm which we only chose for its pedagogical value; more sophisticated RL algorithms are crucial for eliciting high intelligence levels when one is confronted with more complicated learning problems.)

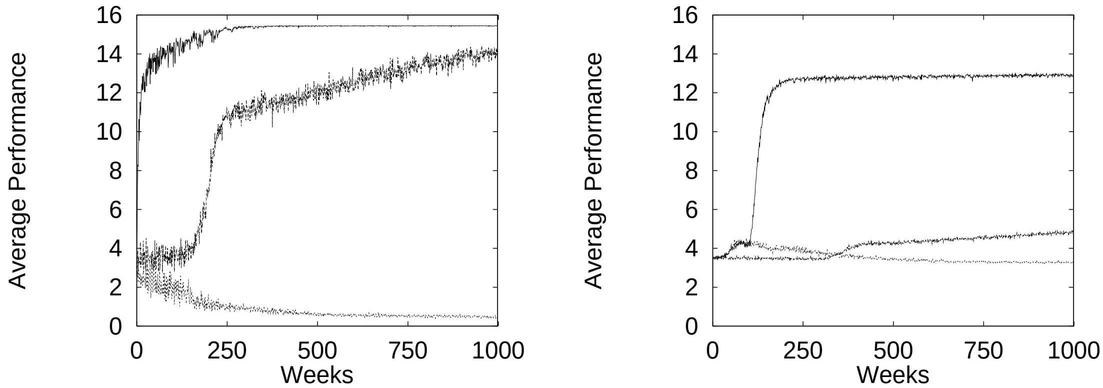

Figure 1: Average world reward when $\vec{\alpha} = [0\ 0\ 0\ 7\ 0\ 0\ 0]$ (left) and when $\vec{\alpha} = [1\ 1\ 1\ 1\ 1\ 1\ 1]$ (right). In both plots the top curve is WL, middle is G, and bottom is UD.

Figure 1 presents world reward values as a function of time, averaged over 50 separate runs, for all three reward functions, for both $\vec{\alpha} = [1\,1\,1\,1\,1\,1\,1]$ and $\vec{\alpha} = [0\,0\,0\,7\,0\,0\,0]$. The behavior with the G reward eventually converges to the global optimum. This is in agreement with the results obtained by Crites [52] for the bank of elevators control problem. Systems using the WL reward also converged to optimal performance. This indicates that for the bar problem our approximations of effects sets are sufficiently accurate, i.e., that ignoring the effects one agent's actions will have on future actions of other agents does not significantly diminish performance. This reflects the fact that the only interactions between agents occurs indirectly, via their affecting each others' reward

with it the system is not factored.

values.

However since the WL reward is more learnable than than the G reward, convergence with the WL reward should be far quicker than with the G reward. Indeed, when $\vec{\alpha} = [0\ 0\ 0\ 7\ 0\ 0\ 0]$, systems using the G reward converge in 1250 weeks, which is 5 times worse than the systems using WL reward. When $\vec{\alpha} = [1\ 1\ 1\ 1\ 1\ 1\ 1]$ systems take 6500 weeks to converge with the G reward, which is more than *30 times* worse than the time with the WL reward.

In contrast to the behavior for COIN theory-based reward functions, use of the conventional UD reward results in very poor world reward values, values that deteriorated as the learning progressed. This is an instance of the TOC. For example, for the case where $\vec{\alpha} = [0\ 0\ 0\ 7\ 0\ 0\ 0]$, it is in every agent's interest to attend the same night — but their doing so shrinks the world reward "pie" that must be divided among all agents. A similar TOC occurs when $\vec{\alpha}$ is uniform. This is illustrated in fig. 2 which shows a typical example of $\{x_k(\underline{\zeta}_{,t})\}$ for each of the three reward functions for $t = 2000$. In this example optimal performance (achieved with the WL reward) has 6 agents each on 6 separate nights, and the remaining 132 agents on one night.

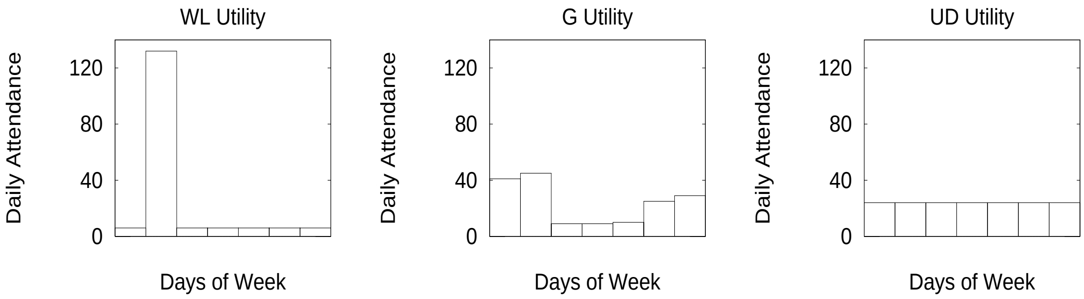

Figure 2: Typical daily attendance when $\vec{\alpha} = [1\ 1\ 1\ 1\ 1\ 1\ 1]$ for WL (left), G (center), and UD (right).

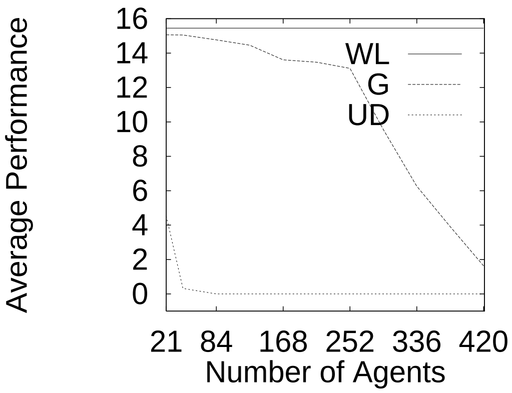

Figure 3: Behavior of each reward function with respect to the number of agents for $\vec{\alpha} = [0\ 0\ 0\ 7\ 0\ 0\ 0]$.

Figure 3 shows how $t = 2000$ performance scales with N for each of the reward signals for $\vec{\alpha} = [0\ 0\ 0\ 7\ 0\ 0\ 0]$. Systems using the UD reward perform poorly regardless of N. Systems using the G reward perform well when N is low. As N increases however, it becomes increasingly difficult for the agents to extract the information they need from the G reward. (This problem is significantly worse for uniform $\vec{\alpha}$.) Because of their superior learnability, systems using the WL reward overcome this signal-to-noise problem: the WL reward is based on clamping the states of all agents but one, and therefore is not affected by the number of agents in the system.

Kagan: What do you mean by that last sentence? Only one agent gets clamped - the agent whose reward is being calculated. Also, is it possible to stretch fig. 3 horizontally, so that the differences in the line types are more visible?

4.5 Macrolearning

In the experiments recounted above the agents' were sufficiently independent that assuming they did not affect each other's actions (when forming guesses for effect sets) allowed the resultant WL reward signals to result in optimal performance. In this section we investigate the contrasting situation where we have initial guesses of effect sets that are quite poor and that therefore result in bad global performance when used with WL rewards. In particular, we investigate the use of macrolearning to correct those guessed effect sets at run-time, so that with the corrected guessed effect sets WL rewards will instead give optimal performance. This models real-world scenarios where the system designer's initial guessed effect sets are poor approximations of the actual associated effect sets and need to be corrected adaptively.

In these experiments the bar problem is significantly modified to incorporate constraints designed to result in poor G when the WL reward is used with certain initial guessed effect sets. To do this we forced the nights actually attended by some of the agents (followers) to agree with those attended by other agents (leaders), regardless of what night those followers "picked" via their microlearning algorithms. (For leaders, picked and actually attended nights were always the same.) We then had the world utility be the sum, over all leaders, of the values of a triply-indexed reward matrix whose indices are the the nights that each leader-follower set attends: $G(\underline{\zeta}) = \sum_t \sum_i R_{l_i(t), f1_i(t), f2_i(t)}$ where $l_i(t)$ is the night the i^{th} leader attends in week t, and $f1_i(t)$ and $f2_i(t)$ are the nights attended by the followers of leader i, in week t (in this study, each leader has two followers). We also had the states of each node be one of the integers $\{0, 1, ..., 6\}$ rather than (as in the bar problem) a unary seven-dimensional vector. This was a bit of a contrivance, since constructions like $\partial_{\underline{\zeta}_{\eta,0}}$ aren't meaningful for such essentially symbolic interpretations of the possible states $\underline{\zeta}_{\eta,0}$. But as elaborated below, it was helpful for constructing a scenario in which guessed effect set WLU results in poor performance, i.e., a scenario in which we can explore the application of macrolearning.

To see how this setup can result in poor world utility, first note that the system's dynamics is what restricts all the members of each triple $(l_i(t), f1_i(t), f2_i(t))$ to equal the night picked by leader i for week t. So $f1_i(t)$ and $f2_i(t)$ are both in leader i's actual effect set at week t — whereas the initial guess for i's effect set may or may not contain nodes other than $l_i(t)$. (E.g., in the bar problem experiments, it does not contain any nodes beyond $l_i(t)$.) On the other hand, G and R are defined for all possible triples $(l_i(t), f1_i(t), f2_i(t))$. So in particular, R is defined for the dynamically unrealizable triples that can arise in the clamping operation. This fact, combined with the leader-follower dynamics, means that for certain R's there exist guessed effect sets such that the dynamics assures poor world utility when the associated WL rewards are used. This is precisely the type of problem that macrolearning is designed to correct.

As an example, say each week only contains two nights, 0 and 1. Set $R_{111} = 1$ and $R_{000} = 0$. So the contribution to G when a leader picks night 1 is 1, and when that leader picks night 0 it is 0, independent of the picks of that leader's followers (since the actual nights they attend are determined by their leader's picks). Accordingly, we want to have a private utility for each leader that will induce that leader to pick night 1. Now if a leader's guessed effect set includes both of its followers (in addition to the leader itself), then clamping all elements in its effect set to 0 results in an R value of $R_{000} = 0$. Therefore the associated guessed effect set WLU will reward the leader for choosing night 1, which is what we want. (For this case WL reward equals $R_{111} - R_{000} = 1$ if the leader picks night 1, compared to reward $R_{000} - R_{000} = 0$ for picking night 0.)

However consider having two leaders, i_1 and i_2, where i_1's guessed effect set consists of i_1 itself together with the two followers of i_2 (rather than together with the two followers of i_1 itself). So neither of leader i_1's followers are in its guessed effect set, while i_1 itself is. Accordingly, the three indices to i_1's R need not have the same value. Similarly, clamping the nodes in its guessed effect set won't affect the values of the second and third indices to i_1's R, since the values of those indices are set by i_1's followers. So for example, if i_2 and its two followers go to night 0 in week 0, and i_1 goes to night 1 in that week, then the associated guessed effect set wonderful life reward for i_1 for week 0 is $G(\underline{\zeta}_{,t=0}) - G(\mathrm{CL}_{l_{i_1}(0), f1_{i_2}(0), f2_{i_2}(0)}(\underline{\zeta}_{,t=0})) = R_{l_{i_1}(0), f1_{i_1}(0), f2_{i_1}(0)} + R_{l_{i_2}(0), f1_{i_2}(0), f2_{i_2}(0)} - [R_{0, f1_{i_1}(0), f2_{i_1}(0)} + R_{l_{i_2}(0), 0, 0}]$. This equals $R_{111} + R_{000} - R_{011} - R_{000} = 1 - R_{011}$. Simply by setting $R_{011} < -1$ we can ensure that this is negative. Conversely, if leader i_1 had gone to night 0, its guessed effect WLU would have been 0. So in this situation leader i_1 will get a greater reward for going to night 0 than for going to night 1. In this situation, leader i_1's using its guessed effect set WLU will lead it to make the wrong pick.

To investigate the efficacy of the macrolearning, two sets of separate experiments were conducted. In the first one the reward matrix R was chosen so that if each leader is maximizing its WL reward, but for guessed effect sets that contain none of its followers, then the system evolves to *minimal* world reward. So if a leader incorrectly guesses that some σ is its effect set even though σ doesn't contain both of that leader's followers, and if this is true for all leaders, then we are assured of worst possible performance. In the second set of experiments, we investigated the efficacy of macrolearning for a broader spectrum of reward matrices by generating those matrices randomly. We call these two kinds of reward matrices *worst-case* and *random* reward matrices, respectively.

In both cases, if it can modify the initial guessed effect sets of the leaders to include their followers, then macrolearning will induce the system to be factored.

The microlearning in these experiments was the same as in the bar problem. All experiments used the WL personal reward with some (intially random) guessed effect set. When macrolearning was used, it was implemented starting after the microlearning had run for a specified number of weeks. The macrolearner worked by estimating the correlations between the agents' selections of which nights to attend. It did this by examining the attendances of the agents over the preceding weeks. Given those estimates, for each agent η the two agents whose attendences were estimated to be the most correlated with those of agent η were put into agent η's guessed effect set. Of course, none of this macrolearning had any effect on global performance when applied to follower agents, but the macrolearning algorithm cannot know that ahead of time; it applied this procedure to each and every agent in the system.

Figure 4 presents averages over 50 of world reward as a function of weeks using the worst-case reward matrix. For comparison purposes, in both plots the top curve represents the case where the followers are in their leader's guessed effect sets. The bottom curve in both plots represents the other extreme where no leader's guessed effect set contains either of its followers. In both plots, the middle curve is performance when the leaders' guessed effect sets are initially random, both with (right) and without (left) macrolearning turned on at week 500.

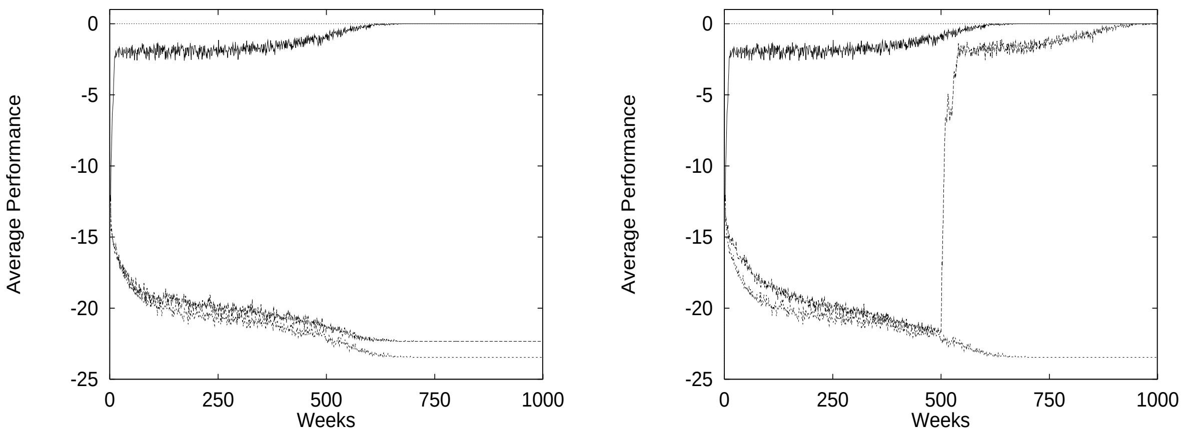

Figure 4: Leader-follower problem with worst case reward matrix. In both plots, every follower is in its leader's guessed effect set in the top curve, no follower is in its leader's guessed effect set in the bottom curve, and followers are randomly asigned to guessed effect sets of the leaders in the middle curve. The two plots are without (left) and with (right) macrolearning at 500 weeks.

The performance for random guessed effect sets differs only slightly from that of having leaders' guessed effect sets contain none of their followers; both start with poor values of world reward that deteriorates with time. However, when macrolearning is performed on systems with initially random guessed effect sets, the system quickly rectifies itself and converges to optimal performance. This is reflected by the sudden vertical jump through the middle of the right plot at 500 weeks, the point at which macrolearning changed the guessed effect sets. By changing those guessed effect sets macrolearning results in a system that is factored for the associated WL reward function, so that those reward functions quickly induced the maximal possible world reward.

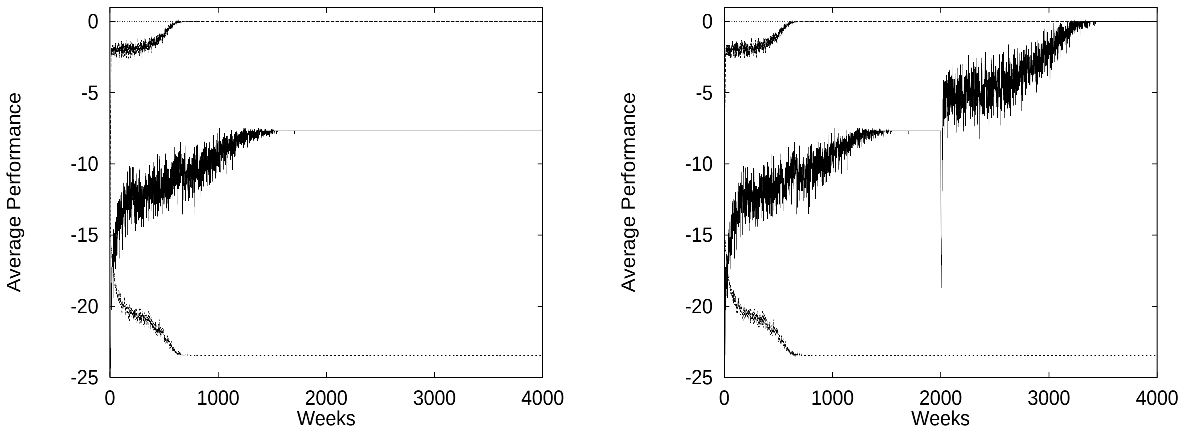

Figure 5: Leader-follower problem for random reward matrices. The ordering of the plots is exactly as in Figure 4. Macrolearning is applied at 2000 weeks, in the right plot.

Figure 5 presents performance averaged over 50 runs for world reward as a function of weeks using a spectrum of reward matrices selected at random. The ordering of the plots is exactly as in Figure 4. Macrolearning is applied at 2000 weeks, in the right plot. The simulations in Figure 5 were lengthened from those in Figure 4 because the convergence time of the full spectrum of reward matrices case was longer.

In figure 5 the macrolearning resulted in a transient degradation in performance at 2000 weeks followed by convergence to the optimal. Without macrolearning the system's performance no longer varied after 2000 weeks. Combined with the results presented in Figure 4, these experiments demonstrate that macrolearning induces optimal performance by aligning the agents' guessed effect sets with those agents that they actually do influence the most.

Acknowledgements: The authors would like to thank Ann Bell, Justin Boyan, Hal Duncan, Robin Morris, Michael New and Joe Sill for their comments and Kevin Wheeler for his help with the simulations.

References

[1] H. Abelson and N. Forbes. Morphous-computing techniques may lead to intelligent materials. *Computers in Physics*, 12(6):520–522, 1998.

[2] M. R. Anderson and T. W. Sandholm. Leveled commitment contracts with myopic and strategic agents. In *Proceedings of the Fifteenth National Conference on Artificial Intelligence*, pages 36–45, 1998.

[3] K. Arrow and G. Debreu. The existence of an equilibrium for a competitive equilibrium. *Econometrica*, 22:265–290, 1954.

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

[5] C. G. Atkenson, S. A. Schaal, and A. W. Moore. Locally weighted learning. *Artificial Intelligence Review*, 11:11–73, 1997.

[6] C. G. Atkeson. Nonparametric model-based reinforcement learning. In *Advances in Neural Information Processing Systems - 10*, pages 1008–1014. MIT Press, 1998.

[7] R. J. Aumann. Correlated equilibrium as an expression of Bayesian rationality. *Econometrica*, 55(1):1–18, 1987.

[8] R. Axelrod. *The Evolution of Cooperation.* Basic Books, NY, 1984.

[9] R. Axelrod. *The Complexity of Cooperation: Agent–Based Models of Competition and Collaboration.* Princeton University Press, NJ, 1997.

[10] P. Bak, S. F. Norrelykke, and M. Shubik. The dynamics of money. 1998.

[11] P. Bak and K. Sneppen. Punctuated equilibrium and criticality in a simple model of evolution. *Physical Review Letters*, 71(24):4083–4086, 1993.

[12] P. Bak, C. Tang, and K. Wiesenfeld. *Physical Review A*, 38:364, 1987.

[13] M. Bando, K. Hasebe, A. Nakayama, A. Shibata, and Y. Sugiyama. Dynamical model of traffic congestion and numerical simulation. *Physical Review E*, 51(2):1035–1042, 1995.

[14] S. Bankes. Exploring the foundations of artificial societies: Experiments in evolving solutions to the iterated N–player prisoner's dilemma. In R. Brooks and P. Maes, editors, *Artificial Life IV*, pages 337–342. MIT Press, 1994.

[15] M. F. Barnsely, editor. *Chaotic Dynamics and Fraactals.* Academic Press, 1986.

[16] T. Bass. Road to ruin. *Discover*, pages 56–61, May 1992.

[17] M. Batty. Predicting where we walk. *Nature*, 388:19–20, July 1997.

[18] E. Baum. Toward a model of mind as a laissez-faire economy of idiots. In L. Saitta, editor, *Proceedings of the 13th International Conference on Machine Learning*, pages 28–36. Morgan Kaufman, 1996.

[19] E. Baum. Manifesto for an evolutionary economics of intelligence. In C. M. Bishop, editor, *Neural Networks and Machine Learning*. Springer-Verlag, 1998.

[20] E. Baum. Toward a model of mind as an economy of agents. *Machine Learning*, 1999. (in Press).

[21] J. bendor and P. Swistak. The evolutionary advantage of conditional cooperation. *Complexity*, 4(2):15–18, 1996.

[22] J. Berg and A. Engel. Matrix games, mixed strategies, and statistical mechanics. preprint cond-mat/9809265, 1998.

[23] D. Bertsekas and R. Gallager. *Data Networks.* Prentice Hall, Englewood Cliffs, NJ, 1992.

[24] O Biham and A. A. Middleton. Self-organization and a dynamical transition in traffic-flow models. *Physical Review A*, 46(10):R6124–6127, 1992.

[25] K. Binmore. *Fun and Games: A Text on Game Theory.* D. C. Heath and Company, Lexington, MA, 1992.

[26] L. E. Blume and D. Easley. Optimality and natural selection in markets. pre-print: econ-wpa 9712003.pdf, 1997.

[27] E. Bonabeau, F. Henaux, S. Guerin, D. Snyders, P. Kuntz, and G. Theraulaz. Routing in telecommunications networks with "smart" and-like agents. pre-print, 1999.

[28] E. Bonabeau, A. Sobkowski, G. Theraulaz, and J.-L. Deneubourg. Adaptive task allocation inspired by a model of division of labor of social insects. pre-print, 1999.

[29] V. S. Borkar, S. Jain, and G. Rangarajan. Collective behaviour and diversity in economic communities: Some insights from an evolutionary game. In *Proceedings of the Workshop on Econophysics*, Budapest, Hungary, 1997.

[30] V. S. Borkar, S. Jain, and G. Rangarajan. Dynamics of individual specialization and global diversification in communities. *Complexity*, 3(3):50–56, 1998.

[31] C. Boutilier. Learning conventions in multiagent stochastic domains using likelihood estimates. pre-print, 1999.

[32] C. Boutilier. Planning, learning and coordination in multiagent decision processes. pre-print, 1999.

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

[34] J. Boyan and M. Littman. Packet routing in dynamically changing networks: A reinforcement learning approach. In J. Cowan, G. Tesauro, and J. Alspector, editors, *Advances in Neural Information Processing Systems - 6*, pages 671–678. Morgan Kaufmann, 1994.

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

[36] J. Breyer, J. Ackermann, and J. McCaskill. Evolving reaction–diffucion ecosystems with self–assembling structures in thin films. *Artificial Life*, 4:25–40, 1998.

[37] R. A. Brooks. Intelligence without reason. In *Proceedings of the Twelveth Intarnationl Joint Conference on Artificial Intelligence*, pages 569–595, 1991.

[38] R. A. Brooks. Intelligence without representation. *Artificial Intelligence*, 47:139–159, 1991.

[39] T. X. Brown, H. Tong, and S. Singh. Optimizing admission control while ensuring quality of service in multimedia networks via reinforcement learning. In *Advances in Neural Information Processing Systems - 11.* MIT Press, 1999.

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

[41] G. A. Carpenter and S. Grossberg. The ART of adaptive pattern recognition by a self-organizing neural network. *IEEE Computer*, 21(3):77–88, 1988.

[42] A. R. Cassandra, L. P. Kaelbling, and M. L. Littman. Acting optimally in partially observable stochastic domains. In *Proceedings of the 12th National Conference on Artificial Intelligence*, 1994.

[43] A. Cavagna. Irrelevance of memory in the minority game. preprint cond-mat/9812215, December 1998.

[44] D. Challet and Y. C. Zhang. Emergence of cooperation and organization in an evolutionary game. *Physica A*, 246(3-4):407, 1997.

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

[46] J. Cheng. The mixed strategy equilibria and adaptive dynamics in the bar problem. Technical report, Santa Fe Institute Computational Economics Workshop, 1997.

[47] D. R. Cheriton and K. Harty. A market approach to operating system memory allocation. In S.E. Clearwater, editor, *Market-Based Control: A Paradigm for Distributed Resource Allocation.* World Scientific, 1995.

[48] S. P. M. Choi and D. Y. Yeung. Predictive Q-routing: A memory based reinforcement learning approach to adaptive traffic control. In D. S. Touretzky, M. C. Mozer, and M. E. Hasselmo, editors, *Advances in Neural Information Processing Systems - 8*, pages 945–951. MIT Press, 1996.

[49] D. J. Christini and J. J. Collins. Using noise and chaos control to control nonchaotic systems. *Physical Review E*, 52(6):5806–5809, 1995.

[50] A. Church. The calculi of lambda-conversion. In *Annals of Mathematics Studies, no. 6*. Princeton University Press, 1941.

[51] 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, June 1998.

[52] 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.

[53] J. P. Crutchfield and J. E. hanson. Turbulent pattern bases for cellular automata. *Physica D*, 69:279–301, 1993.

[54] M. H. New D. Wolpert and Ann Bell. Distorting reward functions to improve reinforcement learning. 1999. (pre-print).

[55] R. Das, M. Mitchell, and J. P. Crutchfield. A genetic algorithm discovers particle-based computation in cellular automata. In Y. Davidor, H.-P. Schwefel, and R. Manner, editors, *Parallel Problem Solving from Nature III*, pages 344–353. Springer-Verlag, 1998.

[56] J. de Boer, B. Derrida, H. Flyvberg, A. D. Jackson, and T. Wettig. *Physical Review Letters*, 73(6):906–909, 1994.

[57] M. A. R. de Cara, O. Pla, and F. Guinea. Competition, efficiency and collective behavior in the "El Farol" bar model. preprint cond-mat/9811162 (to appear in European Physics Journal B), November 1998.

[58] A. de Vany. The emergence and evolution of self-organized coalitions. In M. Gilli, editor, *Comutational Methods in Economics*. Kluwer Scientific Publishers, 1999. (to appear).

[59] D. W. Deamer and J. Oro. The role of lipids in prebiotic structures. *Biosystems*, 12:167–175, 1980.

[60] M. Dorigo and L. M. Gambardella. Ant colonies for the travelling salesman problem. *Biosystems*, 39, 1997.

[61] M. Dorigo and L. M. Gambardella. Ant colony systems: A cooperative learning approach to the travelling salesman problem. *IEEE Transactions on Evolutionay Computation*, 1(1):53–66, 1997.

[62] B. Drossel. A simple model for the formation of a complex organism. preprint adap-org/9811002, November 1998.

[63] A. A. Economides and J. A. Silvester. Multi-objective routing in integrated services networks: A game theory approach. In *IEEE Infocom '91: Proceedings of the Conference on Computer Communication*, volume 3, 1991.

[64] L. Edwards, Y. Peng, and J. A. Reggia. Computational models for the formation of protocell structures. *Artificial Life*, 4:61–77, 1998.

[65] C. M. Ellison. The Utah TENEX scheduler. *Proceedings of the IEEE*, 63:940–945, 1975.

[66] J. M. Epstein. Zones of cooperation in demographic prisoner's dilemma. *Complexity*, 4(2):36–48, 1996.

[67] J. M. Epstein. *Nonlinear Dynamics, Mathematical Biology, and Social Science*. Addison Wesley, Reading, MA, 1997.

[68] J. M. Epstein and R. Axtell. *Growing Artificial Societies: Social Sciences from the Bottom Up*. MIT Press, Cambridge, MA, 1996.

[69] Y. M. Ermoliev and S. D. Flam. Learning in potential games. Technical Report IR-97-022, International Institute for Applied Systems Analysis, June 1997.

[70] N. Feltovich. Equilibrium and reinforcement learning with private information: An experimental study. pre-print, Dept. of Economics, U. of Houston, July 1997.

[71] J. Ferber. Reactive distributed artificial intelligence: Principles and applications. In G. O-Hare and N. Jennings, editors, *Foundations of Distributed Artificial Intelligence*, pages 287–314. Wiley, 1996.

[72] D. F. Ferguson, C. Nikolaou, and Y. Yemini. An economy for flow control in computer networks. In *IEEE Infocom '89*, pages 110–118, 1989.

[73] J. Filar and K. Vrieze. *Competitive Markov Decision Processes*. Springer-Verlag, 1997.

[74] W. Fontana. Algorithmic chemistry. In C. G. Langton, C. Taylor, J. D. Farmer, and S. Rasmussen, editors, *Artificial Life II*, pages 159–209. Addison Wesley, 1992.

[75] C. L. Forgy. RETE: A fast algorithm for the many pattern/many object pattent match problem. *Artificial Intelligence*, 19(1):17–37, 1982.

[76] B. M. Friedman and F. H. Hahn, editors. *The Handbook of Monetary Economics Vol. I.* North-Holland, Amsterdam, 1990.

[77] D. Fudenberg and D. K. Levine. Steady state learning and Nash equilibrium. *Econometrica*, 61(3):547–573, 1993.

[78] D. Fudenberg and D. K. Levine. *The Theory of Learning in Games.* MIT Press, Cambridge, MA, 1998.

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

[80] Gabora. Autocatalytic closure in a cognitive system: A tentative scenario for the origin of culture. *Psycoloquy*, 9(67), December 1998.

[81] V. V Gafiychuk. Distributed self–regulation induced by negative feedbacks in ecological and economic systems. pre-print, adap-org/98110011, November 1998.

[82] S. Galam. Spontaneous coalition forming: A model from spin glass. pre-print cond-mat/9901022, January 1999.

[83] C.W. Gardiner. *Handbook of Stochastic Methods.* Springer-Verlag, NY, NY, 1985.

[84] C. L. Giles, G. M. Kuhn, and R. J. Williams. Dynamic recurrent networks: theory and applications. *IEEE Transactions on Neural Networks*, 5:153–156, March 1994.

[85] C. V. Goldman and J. S. Rosenschein. Emergent coordination through the use of cooperative state–changing rules. pre-print, 1999.

[86] S. J. Gould and N. Eldredge. *Paleobiology*, 3:115, 1977.

[87] O. Guenther, T. Hogg, and B. A. Huberman. Learning in multiagent control of smart matter. In *AAAI-97 Workshop on Multiagent Learning*, 1997.

[88] 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.

[89] E. A. Hansen, A. G. Barto, and S. Zilberstein. Reinforcement learning for mixed open-loop and closed loop control. In *Advances in Neural Information Processing Systems - 9*, pages 1026–1032. MIT Press, 1998.

[90] I Hanski. Be diverse, be predictable. *Nature*, 390:440–441, 1997.

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

[92] D. Helbing, J. Keltsch, and P. Molnar. Modelling the evolution of the human trail systems. *Nature*, 388:47–49, July 1997.

[93] D. Helbing, F. Schweitzer, J. Keltsch, and P. Molnar. Active walker model for the formation of human and animal trail systems. *Physical Review E*, 56(3):2527–2539, 1997.

[94] D. Helbing and M. Treiber. Jams, waves, and clusters. *Science*, 282:200–201, December 1998.

[95] D. Helbing and M. Treiber. Phase diagram of traffic states in the presence of inhomogeneities. *Physics Review Letters*, 81:3042, 1998.

[96] M. Herrmann and B. S. kerner. Local cluster effect in different traffic flow models. *Physica A*, 225:163–168, 1998.

[97] T. Hogg and B. A. Huberman. Controlling smart matter. *Smart Materials and Structures*, 7:R1–R14, 1998.

[98] J. Holland and J. H. Miller. Artificial adaptive agents in economic theory. *American Economic Review*, 81:365–370, May 1991.

[99] J. H. Holland, editor. *Adaptation in Natural and Artificial Systems.* MIT Press, 1993.

[100] J. J. Hopfield. Neural networks and physical systems with emergent collective computational abilities. *Proceedings National Academy of Science*, 79:2554–2558, April 1982.

[101] J. J. Hopfield and David W. Tank. Neural computation of decisions in optimization problems. *Biological Cybernetics*, 52:141–152, 1985.

[102] J. J. Hopfield and David W. Tank. Collective computation with continuous variables. *Disordered Systems and Biological Organization*, 1986.

[103] B. G. Horne and C. Lee Giles. An experimental comparison of recurrent neural networks. In G. Tesauro, D. S. Touretzky, and T. K. Leen, editors, *Advances in Neural Information Processing Systems - 7*, pages 697–704. MIT Press, 1995.

[104] M.-T. T. Hsiao and A. A. Lazar. Optimal flow control of multi-class queueing networks with decentralized information. In *IEEE Infocom '89*, pages 652–661, 1987.

[105] 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.

[106] 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.

[107] M. Huber and R. A. Grupen. Learning to coordinate controllers – reinforcement learning on a control basis. In *Proceedings of the 15th International Conference of Artificial Intelligence*, volume 2, pages 1366–1371, 1997.

[108] B. A. Hubermann, editor. *The Ecology of Computation.* North-Holland, Amsterdam, 1988.

[109] B. A. Hubermann and S. H. Clearwater. A multi-agent system for controlling building environments. In *Proceedings of the International Conference on Multiagent Systems*, pages 171–176, 1995.

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

[111] M. E. Huhns, editor. *Distributed Artificial Intelligence.* Pittman, London, 1987.

[112] R. V. Iyer and S. Ghosh. DARYN, a distributed decision–making algorithm for railway networks: Modeling and simulation. *IEEE transaction of Vehicular Technology*, 44(1):180–191, 1995.

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

[114] 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.

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

[116] E. Kalai and E. Lehrer. Rational learning leads to Nash equilibrium. *Econometrica*, 61(5):1019–1045, 1993.

[117] S. A. Kauffman. *At Home in the Universe: The Search for the Laws of Self-Organization and Complexity.* Oxford University Press, 1995.

[118] L. Keller and H. K. Reeve. Familiarity breeds cooperation. *Nature*, 394:121–122, 1998.

[119] J. O. Kephart. A biologically inspired immune system for computers. In R. Brooks and P. Maes, editors, *Artificial Life IV*, pages 130–139. MIT Press, 1994.

[120] J. O. Kephart, J. E. Hanson, and J. Sairamesh. Price and niche wars in a free-market economy of software agents. *Artificial Life*, 4:1–13, 1998.

[121] B. S. Kerner, P. Konhauser, and M. Schilke. Deterministic spontaneous appearance of traffic jams in slightly inhomogeneous traffic flow. *Physical Review E*, 51(6):6243–6246, 1995.

[122] B. S. Kerner and H. Rehborn. Experimental properties of complexity in traffic flow. *Physical Review E*, 53(5):R4275–4278, 1996.

[123] T. F. Knight and G. J. Sussman. Cellular gate technology. In *Proceedings of the First International Conference on Unconventional Models of Computation*, Auckland, NZ, January 1998.

[124] 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.

[125] S. Kraus. Negotiation and cooperation in multi-agent environments. *Artificial Intelligence*, pages 79–97, 1997.

[126] V. Krishna and P. Motty. Efficient mechanism design. pre-print, 1997.

[127] J. F. Kurose and R. Simha. A microeconomic approach to optimail resource allocation in distributed computer systems. *IEEE Transactions on Computers*, 35(5):705–717, 1989.

[128] M. Kurz. On the structure and diversity of rational beliefs. *Economic Theory*, 4:877–900, 1994.

[129] M. Kurz. Rational beliefs and endogenous uncertainty. *Economic Theory*, 8:383–397, 1996.

[130] R. J. La and V. Anantharam. Optimal routing control: Game theoretic approach. (submitted to IEEE transactions on Automatic Control), 1999.

[131] C. G. Langton. Introduction. In C. G. Langton, C. Taylor, J. D. Farmer, and S. Rasmussen, editors, *Artificial Life II*, pages 3–25. Addison Wesley, 1992.

[132] A. A. Lazar, A. Orda, and D. E. Pendarakis. Capacity allocation under nooncooperative routing. *IEEE Transactions on Networking*, 5(6):861–871, 1997.

[133] A. A. Lazar and N. Semret. Design, analysis and simulation of the progressive second price auction for network bandwidth sharing. Technical Report 487-98-21 (Rev 2.10), Columbia University, April 1998.

[134] T. S. Lee, S. Ghosh, J. Liu, X. Ge, and A. Nerode. A mathematical frmakework for asynchronous, distributed, decision–making systems with semi–autonomous entities: Algorithm sythesis, simulation, and evaluation. In *Fourth International Symposium on Autonomous Decentralized Systems*, Tokyo, Japan, 1999.

[135] T. M. Lenton. Gaia and natural selection. *Nature*, 394:439–447, 1998.

[136] 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.

[137] M. L. Littman and J. Boyan. A distributed reinforcement learning scheme for network routing. In *Proceedings of the 1993 International Workshop on Applications of Neural Networks to Telecommunications*, pages 45–51, 1993.

[138] Ostroy J. M. and R. M. Starr. The transaction role of money. In B. M. Friedman and F. H. Hahn, editors, *The Handbook of Monetary Economics Vol. I.* North-Holland, Amsterdam, 1990.

[139] J. K. MacKie-Mason and R. V. Hal. Pricing congestible network resources. *IEEE Journal on Selected Areas of Communications*, 13(7):1141–1149, 1995.

[140] W. G. Macready and D. H. Wolpert. What makes an optimization problem hard? *Complexity*, 5:40–46, 1996.

[141] W. G. Macready and D. H. Wolpert. Bandit problems and the exploration/exploitatin tradeoff. *IEEE Transactions on Evolutionary Computation*, 2:2–22, 1998.

[142] P. Maes. *Designing Autonomous Agents.* MIT Press, Cambridge, MA, 1990.

[143] P. Marbach, O. Mihatsch, M. Schulte, and J. Tsisiklis. Reinforcement learning for call admission control and routing in integrated service networks. In *Advances in Neural Information Processing Systems - 10*, pages 922–928. MIT Press, 1998.

[144] M. Marsili and Y.-C. Zhang. Stochastic dynamics in game theory. preprint cond-mat/9801309, January 1998.

[145] D. McFarland. Toward robot cooperation. In *From Animals to Animats 3: Proceedings of the Third International Conference on Simulation of Adaptive Behavior*, pages 440–443. MIT Press, 1994.

[146] B. McMullin and F. J. Varela. Rediscovering computational autopoiesis. pre-print SFI Working paper 07-02-012, 1999.

[147] D. A. Meyer and T. A. Brown. Statistical mechanics of voting. *Physics Review Letters*, 81(8):1718–1721, 1998.

[148] S. Milgram. The small world problem. *Psychology Today*, 2:60–67, 1967.

[149] J. H. Miller. The coevolution of automata in the repeated prisoner's dilemma. *Journal of Economic Behavior and Organization*, 29(1):87–112, 1996.

[150] J. H. Miller. Evolving information processing organizations. pre-print, 1996.

[151] J. H. Miller and J. Andreoni. Auctions with adaptive artificial agents. *Journal of Games and Economic Behavior*, 10:39–64, 1995.

[152] J. H. Miller, C. Butts, and D. Rode. Communication and cooperation. pre-print, 1998.

[153] J. Mirrlees. An exploration in the theory of optimal income taxation. *Review of Economic Studies*, 38:175–208, 1974.

[154] M. Mitchell. Computation in cellular automata: A selected review. In T. Gramss, editor, *Nonstandard Computation*. Weinheim, 1998.

[155] M. Mitchell, J. P. Crutchfield, and P. T. Hraber. Evolving cellular automata to perform computations: Mechanisms and impediments. *Physica D*, 75:361–391, 1994.

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

[157] A. W. Moore, C. G. Atkeson, and S. Schaal. Locally weighted learning for control. 1997.

[158] D. E. Moriarty and P. Langley. Learning cooperative lane selection strategies for highways. In *Proceedings of the Fifteenth National Conference on Artificial Intelligence*, Madisson, WI, 1998.

[159] R. Munos and P. Bourgine. Reinforcement learning for continuous stochastic control problems. In *Advances in Neural Information Processing Systems - 10*, pages 1029–1035. MIT Press, 1998.

[160] K. Naigel. Experiences with iterated traffic microsimulations in dallas. pre-print adap-org/9712001, December 1997.

[161] K. Naigel, P. Stretz, M. Pieck, S. Leckey, R. Donnelly, and C. Barrett. TRANSIMS traffic flow characteristics. pre-print adap-org/9710003, October 1997.

[162] J. F. Nash. Equilibrium points in N-person games. *Proceedings of the National Academy of Sciences of the United States of America*, 36(48-49), 1950.

[163] A. Neyman. Bounded complexity justifies cooperation in the finitely repeated prisoner's dilemma. *Economics Letters*, 19:227–230, 1985.

[164] S. I. Nishimura and T. Ikegami. Emergence of collective strategies in a prey-predator game model. *Artificial Life*, 3:243–360, 1997.

[165] M. A. Nowak and K. Sigmund. Evolution of inderect reciprocity by image scoring. *Nature*, 393:573–577, 1998.

[166] S. Olafsson. Games on networks. *Proceedings of the IEEE*, 85(10):1556–1562, 1997.

[167] A. Orda, R. Rom, and N. Shimkin. Competitive routing in multiuse communication networks. *IEEE/ACM Transactions on Networking*, 1(5):510–521, 1993.

[168] J. Oro, E. Sherwood, J. Eichberg, and D. Epps. Formation of phospholipids under pritive earth conditions and the role of membranes in prebiological evolution. In *Light Transducing Membranes, Structure, Function and Evolution*, pages 1–21. New York, 1978.

[169] B. A. Pearlmutter. Learning state space trajectories in recurrent neural networks. *Neural Computation*, 1(2):263–269, 1989.

[170] L. A. Pipes. An operational analysis of traffic dynamics. *Journal of Applied Physics*, 24(3):274–281, 1953.

[171] G. A. Polls. Stability is woven by complex webs. *Nature*, 395:744–745, 1998.

[172] M. Potters, R. Cont, and J.-P. Bouchaud. Financial markets as adaptive ecosystems. preprint cond-mat/9609172 v2, June 1997.

[173] D. Prokhorov and D. Wunsch. Adaptive critic design. *IEEE Transactions on Neural Networks*, 8(5):997–1007, 1997.

[174] Z. Qu, F. Xie, and G. Hu. Spatiotemporal on-off intermittency by random driving. *Physical Review E*, 53(2):R1301–1304, 1996.

[175] T. S. Ray. An approach to the synthesis of life. In C. G. Langton, C. Taylor, J. D. Farmer, and S. Rasmussen, editors, *Artificial Life II*, pages 371–408. Addison Wesley, 1992.

[176] T. S. Ray. Evolving multi–cellular artificial life. In R. Brooks and P. Maes, editors, *Artificial Life IV*, pages 283–288. MIT Press, 1994.

[177] T. S. Ray. Evolution, complexity, entropy and artificial life. *Physica D*, 1995.

[178] E. Rich and K. Knight. *Artificial Intelligence*. McGraw-Hill, Inc., 2 edition, 1991.

[179] D. E. Rumelhart and J. L. McClelland, editors. *Parallel Distributed Processing: Explorations in the Microstructure of Cognition*. Bradford Books/MIT Press, Cambridge, MA, 1986.

[180] A. Samuel. Some studies in machine learning using the game of checkers. *IBM Journal of Reseach and Development*, 3:210–229, 1959.

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

[182] 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.

[183] T. Sandholm and V. R. Lesser. Coalitions among computationally bounded agents. *Artificial Intelligence*, 94:99–137, 1997.

[184] R. Savit, R. Manuca, and R. Riolo. Adaptive competition, market efficiency, phase transitions and spin-glasses. preprint cond-mat/9712006, December 1997.

[185] 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.

[186] J. Schmidhuber, J. Zhao, and N. N. Schraudoiph. reinforcement learning with self-modifying policies. In S. Thrun and L. Pratt, editors, *Learning to Learn*. Kluwer, 1997.

[187] R. Schoonderwoerd, O. Holland, and J. Bruten. Ant-like agents for load balancing in telecommunication networks. In *Autonomous Agents 97*, pages 209–216. MIT Press, 1997.

[188] M. Schreckenberg, A. Schadschneider, K. Nagel, and N. Ito. Discrete stochastic models for traffic flow. *Physical Review E*, 51(4):2939–2949, 1995.

[189] J. Schull. Are species intelligent? *Behavioral and Brain Sciences*, 13:63–108, 1990.

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

[191] S. Sen, M. Sekaran, and J. Hale. Learning to coordinate without sharing information. pre-print, 1999.

[192] R. Sethi. Stability of equilibria in games with procedural rational players. pre-print, Dept of Economics, Columbia University, November 1998.

[193] S. J. Shenker. Making greed work in networks: A game-theoretic analysis of switch service disciplines. *IEEE Transactions on Networking*, 3(6):819–831, 1995.

[194] 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.

[195] J. Sidel, P. M. Aoki, S. Barr, A. Sah, C. Staelin, M. Stonebreaker, and Yu A. Data replication in mariposa. In *Proceedings of the 12th International Conference on Data Engineering*, 1996.

[196] S. Sinha and N. Gupte. Adaptive control of spatially extended systems: targeting spatiotemporal patterns and chaos. *Physical Review E*, 58(5):R5221–5224, 1998.

[197] A. R. Smith. Simple nontrivial seft–reproducing machines. In C. G. Langton, C. Taylor, J. D. Farmer, and S. Rasmussen, editors, *Artificial Life II*, pages 709–725. Addison Wesley, 1992.

[198] W. Stallings. *Data and Computer Communications.* MacMillian Publishing Co., New York, 1994.

[199] R. M. Starr. *General Equilibrium Theory.* Cambridge University Press, Cambridge, UK, 1997.

[200] R. M. Starr and M. B. Stinchcombe. Exchange in a network of trading posts. In K. J. Arrow and G. Chichilnisky, editors, *Markets, Information and Uncertainty: Essays in Economic Theory in Honor of Kenneth Arrow.* Cambridge University Press, 1998.

[201] M. Stonebreaker, P. M. Aoki, R. Devine, W. Litwin, and M. Olson. Mariposa: A new architecture for distributed data. In *Proceedings of the 10th International Conference on Data Engineering*, 1994.

[202] D. Subramanian, P. Druschel, and J. Chen. Ants and reinforcement learning: A case study in routing in dynamic networks. In *Proceedings of the Fifteenth International Conference on Artificial Intelligence*, pages 832–838, 1997.

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

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

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

[206] G. Szabo and C. Toke. Evolutionary prisoner's dilemma game on a square lattice. *Physical Review E*, 58(1):69–73, 1998.

[207] G. Tesauro. Practical issues in temporal difference learning. *Machine Learning*, 8:33–53, 1992.

[208] P. Tucker and F. Berman. On market mechanisms as a sofware techniques. Technical Report CS96–513, University of California, San Diego, December 1996.

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

[210] J. von Neuman. *The Theory of Self–Replicating Automata (Ed. Burks, A. W.).* University of Illinois Press, Urbana, IL, 1966. (Work performed by J. von Neuman in 1952-53).

[211] C. A. Waldspurger, T. Hogg, B. A. Huberman, J. O. Kephart, and W. S. Stornetta. Spawn: A distributed computational economy. *IEEE transactions of Software engineering*, 18(2):103–117, 1992.

[212] J. Walrand and P. Varaiya. *High–Performance Communication Networks*. Morgan Kaufmann, San Fransisco, CA, 1996.

[213] L. Walras. *Elements d'Economie Politique pure*. L. Corbaz, Lausanne, 1874. (translated by W. Jaffe as *Elements of Pure Economics*, Homewood, IL", 1954).

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

[215] D. J. Watts and S. H. Strogatz. Collective dynamics of 'small world' networks. *Nature*, 393:440442, 1998.

[216] R. Weiss, G. Homsy, and R. Nagpal. Programming biological cells. In *Proceedings of the 8th International Conference on Architectural Support for Programming Languages and Operating Systems*, San Jose, NZ, 1998.

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

[218] M. P. Wellman. A computational market model for distributed configuration design. In *Proceedings of the 12th National Conference on Artificial Intelligence*, 1994.

[219] D. Wolpert. Collective intelligence. 1999. (pre-print).

[220] D. Wolpert. A mathematics of bounded rationality. 1999. (pre-print).

[221] D. Wolpert, K. Tumer, and J. Frank. Using collective intelligence to route internet traffic. In *Advances in Neural Information Processing Systems - 11*. MIT Press, 1999.

[222] D. H. Wolpert and W. G. Macready. An empirically observable measure of complexity. In *New England Complex Systems Institute Inaugural Conference Proceeding*. in press.

[223] D. H. Wolpert and W. G. Macready. No free lunch theorems for optimization. *IEEE Transactions on Evolutionary Computation*, 1(1):67–82, 1997.

[224] S. Wright. The roles of mutation, inbreeding, crossbreeding and selection in evolution. *Proceedings of the XI International Congress of Genetics*, 8:209–222, 1932.

[225] H. P. Young. The evolution of conventions. *Econometrica*, 61(1):57–84, 1993.

[226] E. Zambrano. Rationalizable bounded rational behavior. pre-print, 1999.

[227] D. H. Zanette and A. S. Mikhailov. Coherence and clustering in ensembles of neural networks. pre-print, 1999.

[228] W. Zhang and T. G. Dietterich. Solving combinatorial optimization tasks by reinforcement learning: A general methodology applied to resource-constrained scheduling. *Journal of Artificial Intelligence Reseach*, 1991.

[229] Y. C. Zhang. Modeling market mechanism with evolutionary games. preprint cond-mat/9803308, 1998.

[230] G. Zlotkin and J. S. Rosenschein. Coalition, cryptography, and stability: Mechanisms for coalition formation in task oriented domains. pre-print, 1999.