Even Adversarial Agents Should Appear to Agree

Robin Hanson
Sterling Software
Artificial Intelligence Research Branch
NASA Ames Research Center, Mail Stop 244-17
Moffett Field, CA 94035, USA
hanson@charon.arc.nasa.gov
415-604-3361

June 5, 2003

Descriptors: coordination, autonomy, actions, beliefs

Abstract

Distributing authority among autonomous agents can induce inconsistency costs if the agents act as if they disagree. If we define an agent's "marginal beliefs" to be the odds at which it is willing to make bets, we find that a betting market can induce agents to act as if they almost agree, not only with respect to the bets they offer but also other actions they take. In a particular "Mars mining" scenario, I explicitly show how utility maximizing agents, who are autonomous and hence distrust each other, can discover a common consensus and take concrete physical actions as if they agreed with that consensus, lowering costs to the group as a whole. Though limited, the approach has also has many unexplored possibilities.

A previous version of this paper was presented at the IJCAI-91 Workshop on Reasoning in Adversarial Domains

(This version is missing some cites and the graphics.)

1 Introduction

Most distributed AI systems, such as [4], distribute processing, action, and knowledge but not authority [**?**]. Distributed authority, where autonomous agents can choose whether or not to cooperate, is often treated as an unfortunate feature of a problem domain, sometimes unavoidable but to be avoided when possible. But in fact, distributed authority offers enough advantages that we might consider including it even when we could choose to force all the agents in our system to cooperate. When agents treat each other skeptically, agent boundaries provide firewalls against stupidity, so that faulty inferences need not be propagated beyond these boundaries. Distributed authority also allows a diversity of competing humans and human organizations to participate in the system design, with each group contributing their own agents. Poorly designed agents should be out-competed, and hence reflected less and less in total system performance. Possible disadvantages of distributed authority include the overhead for agents to reason about suspecting and negotiating with each other, and the costs of inconsistency, which are the focus of this paper.

That inconsistency is to be avoided, all else being equal, is a standard conclusion of both logical and decision theoretic models of "rational" agents. For example, we expect you probably would not reach your personal goals as efficiently if you acted with one set of beliefs during even hours of the day, and another set of beliefs during odd hours, rather than some intermediate set of beliefs during all hours. Or imagine we built a computer agent containing a vision module and a motor control module, each of which held beliefs about the type and location of nearby objects. Unless communication and processing costs were prohibitive, we would want these two modules to share information, so that they came to roughly agree about what objects were nearby. Yes, sometimes we might want our agent's reasoning to split into different "disagreeing" components, each exploring a different possible hypothetical world. But when our agent takes external *actions*, we prefer these actions to be based on a single consistent set of beliefs; it won't do to have the vision system carefully studying an obstacle dead ahead that the motor system doesn't believe in.

Inconsistency can also cost in a multi-agent system. Imagine that we want to solve a particular problem, and can choose between implementing a centralized system, where a central authority manages inference across the whole system, and a decentralized system, where autonomous agents can each choose their own beliefs and actions. The centralized system could in principle maintain consistency across all its subsystems. But if these subsystems were instead autonomous, they might distrust each other, and so be reluctant to share information and suspicious of what they are told. If these distrusting agents acted on different beliefs, each based on less information, they might perform worse than the centralized alternative.

The purpose of this paper is point out a simple and general mechanism which can often, at a modest cost per question, largely eliminate costs of inconsistency in systems with distributed authority. This mechanism is purely voluntary, and requires only the communication of bits. It is already known to the economics community [8], and is here reformulated for an AI audience. While market-based, it is not subsumed by other market-based approaches to distributed computation, such as bidding for needed resources [9] or selling services [10].

The proposed mechanism should induce autonomous agents to share information, allow them to trust what is so shared, and induce them to act as if they (almost) agree with a set of explicit consensus beliefs. The mechanism does not, however, require that the agents actually come to agree (as in [12]), or that they explicitly agree to act in accordance with either a group plan or any explicit function of what each agent claims to believe (as in [1, 5]). Each disagreeing agent wants to participate in the mechanism; if the system designers had not introduced the mechanism the agents would want to do so themselves. The mechanism does require that the agents be able to state a specific claim about which they disagree, and that this claim be a matter of fact which might be resolved to everyone's satisfaction with sufficient time and effort.

In this paper I describe a concrete situation in which the inconsistency problem arises, propose how to deal with the problem in this context, describe an implementation and simulation of this solution, and then return to a general discussion of the limitations and promise of the mechanism.

2 The Mars Mining Problem

Imagine we work at the Mars Mining company, whose purpose in life is to pick up gold nuggets which are conveniently strewn across the plains of Mars. Since human labor is expensive, we naturally turn to robots. Each day we transport a group of robots to a new location and release the robots to forage in the area, and bring back their loads to the temporary shelter.

A straightforward approach to this problem would create a central controller to manage each group of robots. But let us instead consider a system design which, in addition to distributing action and knowledge, also distributes authority. Each robot is programmed by a different team on earth, is paid for the gold it brings back to the shelter each day, and must pay for expenses like fuel and robot maintenance. This approach allows a great diversity of robot strategies to be tried, but risks situations like the one illustrated in Figure 1.

Figure 1: Mars mining scenario

> Group Delta found a rich area of nuggets one afternoon. But soon afterward, Robot J, scavenging off on a distant hill, radios to the others that he sees a dust storm coming. If he is right, then robots close to the shelter should run for it, to

> avoid the risk of damage during the storm. But they hesitate. Is there really a dust storm, or is Robot J trying to trick his competition out of the gold? What should the robots believe? Should they stay or run?

Figure 2 shows a distribution of ten robots, labeled A though J, each a different distance from the shelter and each with a different private degree of belief in the storm (the result of some unspecified reasoning). If these robots act independently and are equal in all other respects, then whether they run for shelter or not should just be a function of these two parameters. The two shaded regions in Figure 2 distinguish such agents who should run from those who should stay (as determined by an explicit model described in section 5). Robots close to shelter who believe in the storm should run, while close robots who don't believe shouldn't bother. Distant robots should stay no matter what their beliefs, as their chance of escaping the storm by running is much less. Robots B, D, and G should clearly run, and I and J have a slight preference to run.

If we had instead built our Mars mining system around a central controller, it might have forced the system to follow a single set of beliefs. Since the central belief would be followed by all robots, all other possible beliefs would be irrelevant. For example, a central belief in a 47% chance of storm, as shown in Figure 3, would imply a decision boundary of about 880 meters[1]. All robots closer than 880m (A, B, C, D, and E) would run, and the others would stay.

The cost of inconsistency shows itself in situations, like in Figure 2, where a closer autonomous robot like A stays and a further robot like G runs. If these robots could just trade their beliefs, the system as a whole would be better off. The system would have no less chance of finding gold, and a greater chance of avoiding storm damage. If the agents acted as if they agreed, this would not happen.

3 Apparent Beliefs

What does it mean to say that a certain mechanism can make agents "act as if they agree" even when they "really" disagree? Imagine that autonomous agents can maintain a certain level of privacy, hiding internal data structures and financial holdings from prying eyes, and encrypting selected communication. Then the agents "act as if they almost agree" if external observers, trying to infer each agent's beliefs from its observable actions, find those actions consistent with the agents having very similar beliefs [2].

How might external observers attempt such inference? In the example above, if we knew everything about an agent except its belief in the storm, we could infer something about that belief from whether it ran or stayed. But in general, such action data is sparse, underconstraining the inference to beliefs, and indirect, since each action can be influenced by thousands of beliefs.

A more direct and flexible way to probe an agent's beliefs on a particular question is to offer to bet with them on the question. In financial lingo, this is called "buying or

[1] The regions of figures 2 and 3 are slightly different because the central planner can pool assets, and so is less risk-adverse about harm to individual robots.

[2] While a high level of mechanism activity might let one infer that there must be some substantial disagreements somewhere, the agents still "act as if they almost agree" if observers can't figure out who disagrees and on which side.

selling a contingent asset". A "bank" trusted by both parties could issue script like "Worth \$1 *IfStorm*" and "Worth \$1 *IfNotStorm*", and buy or sell pairs like these for exactly \$1. If the base asset, here \$1, can be considered "riskless", and an agent were risk-neutral or had no stake in the question (utility the same whether the claim is true or false) and faithfully followed the axioms of standard decision theory, then the agent's real belief would lie somewhere between the price at which it is willing to buy and sell small amounts of assets contingent on that claim [7].

Of course it is is not computationally possible to exactly follow these axioms; more realistic agents often have stakes in questions and are inconsistent. But because private communication and finances allows agents to make bets in secret, external observers will find it difficult to infer that the agent's net stake in a question is much different from anyone else's. Instead, we can simply define an agent's "marginal belief" in a claim to be somewhere between the mentioned buy and sell prices. If an agent is willing to buy "\$1 *IfStorm*" for \$0.38, and sell it for \$0.40, we say that its marginal belief in the storm is between 38% and 40%. When these buy/sell price ranges collapse to a point, we say have learned the agent's marginal belief exactly; otherwise we have made a coarser measurement. When the buy/sell ranges of different agents overlap, we say they "appear to agree"; when they almost overlap, they "almost appear to agree".

It turns out that if an agent reveals marginal beliefs that violate the standard probability axioms, then any other agent who notices that violation can use it to make a riskless profit, valuable no matter which claims are vindicated. Thus agents should try to avoid easily detectable violations. And, because risk and net stake influences both ordinary and betting actions in the same way, marginal beliefs will also be revealed in ordinary actions, such as whether the robots run or stay. Thus we can more generally declare an agent's marginal beliefs to be those that come closest to implying all of that agent's small actions[3], not just betting actions, under either the no-stakes or the risk-neutral assumption within an ideal decision theory analysis.

4 Betting Markets

Imagine that an external observer, attempting to discover robot A's belief in the storm, approaches A and offers to buy "\$1 *IfStorm*". If A thinks the chance of a storm is 10%, and is alone, then we might expect it to sell "\$1 *IfStorm*" to the observer for near \$0.10, thereby revealing it's belief. But if there is another robot, say B, publicly offering to buy this asset for \$0.38, A will *not* sell it to the observer for \$0.10 – he'd rather get \$0.38 from B. In general, if agents can bet with each other, we would expect them to do so until they do not want to bet anymore, at which point their buy/sell ranges should overlap. At this point an observer could not then tell that their marginal beliefs disagreed – they would appear to agree [**?**]!

Of course if the agent's offer ranges are wide, we can't say we really knew much about their beliefs. But for a small cost one can introduce a simple automatic "broker" on a question, and thereby induce the agents to offer narrow price ranges. Such a broker is simply an agent who always has open buy and sell offers, with a very narrow spread between

[3] It is not clear whether large actions, those affecting a significant fraction of an agent's wealth, could allow true beliefs to be glimpsed behind the veil of marginal beliefs.

the buy and sell price, and a simple safe algorithm for changing those offers as a function of previous sales [2]. Other agents should then find little risk and a potential profit from offering ranges overlapping with and not much wider than the broker's range; they could turn around and sell any unwanted assets to the broker at a profit.

The finite cost of completing a transaction limits how narrow the broker can make its offer ranges, and together with finite communication delays might allow observer to find small discrepancies between market (i.e., broker) and individual marginal beliefs. But such discrepancies can be small since the basic limiting cost is that of sending bits to describe the transaction; all trading can be done electronically, since even money can be sent in bits [3].

Betting markets may make agents appear to agree regarding their betting actions, it why should they appear to agree regarding other actions, like whether the robots in the example above should stay or run from the storm? Because bets can be used as insurance. If without bets a risk-adverse robot would stay, not believing much in the storm, then if she bets enough against the storm she may find it in her interest to run anyway as insurance. If she runs she wins in either case; otherwise all her eggs are in one basket.

If there is enough interest (intrinsic stakes or disagreements) in a question, agents should volunteer to be brokers, as there is money to be made doing so. It is easier for agents to deal with central markets than to search for compatible traders, and most agents would prefer to just take the market price, rather than choose prices they are willing to make offers at. The broker makes money on each "round trip", a buy followed by a sell or vice-versa, and only loses money when he accumulates a net stake on either side of the issue. The simplest broker strategy is to offer prices as a monotonic function of the broker's net stake, i.e., keep making whatever customers are buying more expensive. Competition between brokers should keep the price spreads narrow, and prevent the system from having a single point of failure.

5 Simulation

To demonstrate that this approach can feasibly coordinate simple agents with limited information and computational abilities in a reasonable time, the Mars mining scenario described above was implemented in a short Commonlisp program.

At the beginning of the simulation, each robot had the beliefs and distances specified in Figure 2, and would have chosen the actions implied there in the absence of interaction with the other robots. Instead, a broker posted buy and sell offers, and each robot then repeatedly decided whether their expected utility would be better if they bet some small amount on either side and/or changed their mind about whether to run or stay. The simulation stopped when no one wanted to trade anymore. At that point all the robots acted as if they agreed with the market consensus of 47%, as shown in Figure 3. Not only did they all offer the same betting odds, but everyone who the consensus says should run ran, and those who the consensus says should stay stayed. No matter what the "real" chance of a storm is, this result turns out to be clearly better (higher expected utility) for the system as a whole than if the robots had acted on their initial inclinations. To make the scenario concrete, a variety of arbitrary choices were made. However, I made no search in a space of possible choices to find nice results – all choices are the initial arbitrary choices.

The simulation details are as follows. The agent beliefs and distances are those given in

Figures 2 and 3. Each agent initially has assets consisting of \$1000 cash, robot hardware worth \$1000, the value of the gold that agent expects to pick up if it doesn't run, and initially zero conditional assets (betting contracts). Without a storm there is a 35% chance to find \$1 worth of gold, a 40% chance of \$10, a 20% chance of \$100, and a 5% chance to find \$1000. With a storm each dollar amount increases by 10% [4]. If there is a storm and a robot stays, they are sure to be caught in it. But if they run their chances of evading the storm are $\exp(-\text{distance}/1000\text{m})$, decreasing with distance. If an agent gets caught in the storm, there is a 20% chance of passing unscathed, a 60% chance of suffering 30% damage to the robot hardware, and a 20% chance of needing to replace the hardware[5].

Each robot maximizes expected utility, with utility being the logarithm of an agent's total net worth. Robots consider the current market price as relevant information about whether there will be a storm; they heuristically compute their degree of belief in the storm as a weighted combination of what their private information suggests (weighted 70%), and what the market suggests. This heuristic ignores the price history and the information contained in whether other robots are running to shelter.

Each robot repeatedly cycled through three steps: sending in orders to trade, receiving a reply with the new market price, and computing new orders. To deal with the fact that prices can change during the cycle, each agent sends in multiple orders, each conditional on what the current market price will be when the order arrives. The total cycle takes a random time delay, averaging around 7msec but more for more distant agents.

There is one broker, who initially offers to buy up to \$10 of *IfStorm* at a price of 49%, i.e., \$0.49 per "\$1 *IfStorm*", and sell it at 51%. The broker initially offered the same prices for *IfNotStorm*. If someone bought \$5 of *IfNotStorm*, the prices would remain valid for up to the remaining \$5. When the offer was depleted, the price on *IfNotStorm* would rise to $50 - 52\%$, and the prices on *IfStorm* would drop to $48 - 50\%$. If another \$10 worth were bought, the price would continue on to $51 - 53\%$, etc. If \$10 of *IfStorm* were bought instead, however, the price would go back up to where it started, and the market maker would have made a clear profit of \$0.10.

Figure 5 shows the buy and sell prices as a function of time (in units of simulated seconds). The broker started with assets of \$122.50 [6], lost about half that as the price changed quickly in the first 0.1 time unit, and then profited steadily as the price changed more slowly, with final assets over \$190. Figure 7 shows the stakes held by each player, which, like each agent's beliefs, which is private information not available to the other players. Tick marks indicate when a player changed its mind about whether to run or stay.

Since the final market price of *IfStorm* was $46 - 48\%$, 47% was used as the central belief for calculating the central plan. To avoid a full combinatorial analysis of all possible events, the central run/stay advice was approximated as what an individual would do if they believed the consensus and had the full cash of the group available to reduce risk. This neglected small effects like being able to share the risks associated with how much gold each agent will collect. At 47% the resulting run/stay cutoff was 876m.

To evaluate whether the system as a whole was better off after betting, I calculated

[4] Similar results come without this feature

[5] This is not "death"; they just have to buy a new "body".

[6] The broker could have risked only \$12.25 by offering only \$1 at each price. In the absence of competing brokers offering a better deal, he would have made about the same profit, but trading might have taken ten times longer.

a total expected utility before and after betting, by summing the expected utility of each robot. I calculated this for different possible values of the "true" probability of a storm, and it turns out that no matter what this probability is, the post-betting situation is better than the pre-betting one. This way of calculating total utility is crude, but conservative; it penalizes the final situation for betting risks imposed on agents, without rewarding it for the fact that disagreeing agents each expect to win on average.

While this "betting helps" result should be typical, one can construct situations in which the pre-betting situation is better. If there were only robots A, B, C, E, F, and H, with a consensus price of 14%, and the "true" probability of a storm were $37H$ should run. In this case a divergence of opinion is better, since then at least some of the agents (on his own, B will run) will do the right thing. This scenario is unusual in that everyone with a belief lower than the consensus should do the same as the consensus says, namely to stay; only beliefs substantially higher than the consensus indicate that one should take a different action, namely to run. More typically, acting on beliefs even farther from "true" than the consensus should cause even more harm.

Without a model of how beliefs come to be distributed it is hard to say much about how good the consensus belief is, compared to the individual beliefs, as an estimate of some "true" appropriate belief. The fact that acting on the consensus is better in this "random" situation, and that it takes some work to come up with a similar counter example, offers at least some support for the speculation that it is better on average for system to act as if they agree with such a consensus.

6 Discussion

Having seen the mechanism work in a particular context we are now in a better position to understand its limitations. It requires that agents can express and communicate a common claim to bet on, that there potentially be enough eventual convergence of opinion to uncontroversially settle a bet[7]. If agents can influence the claim bet on, then there can be "moral hazard" with beliefs conditional on who has bet how much. Agent's need to care what happens at the future time when they think they will be vindicated, and interest in a question should last much longer than the communication delays. And the agent's (or some external observer's) interest in the question must be enough to overcome the basic costs [6] of sending messages, carefully working a resolvable claim, and having agents create and search indexes of possibly relevant claims. While it may be in the interest of each agent to act as if they almost agree with the consensus, this does not mean it will always be computationally feasible to do so.

On the other hand, the basic mechanism has many unexplored possibilities. In the scenario above, a smarter broker could have used technical trading techniques to dramatically sped the convergence to consensus. An external user could induce the agent's to answer a question of interest by subsidizing a simple broker who gives away assets to whoever moves the price in the direction of its final resting place. The market prices represent a consensus which autonomous agents with different knowledge could observe and update in parallel, and thus be a "blackboard" [11] for distrusting agents. All logical and conditional statements combining the claims available to bet on imply specific trading strategies which should make

[7] Methods for dealing with this and other problems are discussed in [**?**]

money if those combinations are correct, without exposing such traders to risk regarding other issues. For example, a Bayes network work could be computed by having separate agents each trade on the local constraints expressed by each link in the network. Agents who do not think they know something special about some particular subject can just take the market price as information, setting their personal belief to be that of the consensus. Profits available from arbitrage allow the total system to be more consistent than any individual can afford to be, even in the presence of large numbers of irrational participants.

In conclusion, adversarial agents need not incur the full costs of inconsistency in the presence of a mechanism which makes them act as if they almost agree, even though they really disagree with and distrust each other. Betting markets provide such a mechanism. Simple heuristics allow computationally limited agents to use this mechanism, as demonstrated by the "Mars mining" simulation described. The approach has some clear limitations, but seems a promising area for exploration.

References

[1] M. Bayarri and M. DeGroot. Gaining weight: A Bayesian approach. In Bernardo, DeGroot, Lindley, and Smith, editors, *Bayesian Statistics 3*, pages 25–44. Oxford University Press, 1988.

[2] F. Black. Towards a fully automated exchange. *Financial Analyst Journal*, July and November 1971.

[3] David Chaum. Security without identification: Transaction systems to make big brother obsolete. *Communications of the ACM*, 28(10):1030–1044, October 1985.

[4] D. Durfee and T. Monontgomery. A hierarchical protocol for coordinating multiagent behaviours. In *Eighth National Conference on Artificial Intelligence*, pages 86–93, Menlo Park, 1990. AAAI Press.

[5] C. Genest and J. Zidek. Combining probability distributions: A critique and annotated bibliography. *Statistical Science*, 1(1):114–148, 1986.

[6] Jack Hirshleifer. The private and social value of information and the reward to inventive activity. *American Economics Review*, 61(4):561–74, September 1971.

[7] J. Kadane and R. Winkler. Separating probability elicitation from utilities. *Journal of the American Statistical Association*, 83(402):357–363, June 1988.

[8] J.J. Laffont. *The Economics of Uncertainty and Information.* MIT Press, 1989.

[9] T. Malone, R. Fikes, and M. Howard. Enterprise: A market-like task scheduler for distributed computing environments. In B. Huberman, editor, *The Ecology of Computation*, pages 177–205. North Holland Publishing Company, Amsterdam, 1988.

[10] Mark Miller and K. Eric Drexler. Markets and computation: Agoric open systems. In B. Huberman, editor, *The Ecology of Computation.* North Holland Publishing Company, Amsterdam, 1988.

[11] Penny Nii. Blackboard system: The blackboard model of problem solving and the evolution of blackboard architectures. *AI Magazine*, pages 38–53, Summer 1986.

[12] J. Sebenius and J. Geanakoplos. Don't bet on it: Contingent agreements with asymmetric information. *Journal of the American Statistical Association*, 78(382):424–426, 1983.