Cognitive Instance-Based Learning Agents in a Multi-Agent Congestion Game

Paul Scerri and David Reitter
Carnegie Mellon University and Pennsylvania State University

ABSTRACT

Simultaneous learning by multiple agents can lead to undesirable and unwanted dynamics because the learning creates a non-stationary environment for the other agents to learn against. Although humans often face this challenge, humans can often converge to good cooperative solutions in reasonable amounts of time. In this paper, we empirically compare a multi-agent learning approach inspired by human ways of learning against a more numerically intensive agent way of learning. Specifically, agents must repeatedly traverse a graph and agents on the same edge will interfere with one another so learning is required to find uncongested routes. We find that human inspired instance-based learning performs at least as well as more quantitative and communication intensive approaches. Even when the overall system performance is closely matched, a look at the details of how the result was achieved shows considerable differences between approaches. Some of the advantage can be attributed to the instance-based learning's preference for sticking with known good solutions, because this creates stability that allows the other agents to learn. However, external disturbances, changes to the underlying system, can be more problematic for instance-based learning precisely because so much history is used.

1. INTRODUCTION

Many interesting domains require that robots or agents simultaneously learn and interact with one another for their mutual benefit over time. When the actions of one agent impact the outcomes of another agent, individual learning often leads to complex system dynamics. A canonical example of this problem is cooperative path planning[8, 1], where agents using the same routes negatively interfere with one other, but many other domains have been studied including soccer[16] and markets[27].

One of the core problems in multi-agent learning is that when one agent's behavior may impact the outcome for another agent. When all the agents simultaneously learn, the collective behavior and individual rewards can vary wildly and unpredictably. Controlling these system dynamics in a way that efficiently or even eventually has good behavior has received much attention. Typically, the approach is to somehow change local learning so that undesirable system effects are damped[7, 21]. For example, some agents can hold their behavior fixed while others learn or agents can be made to learn slowly to provide a more stable environment for the other agents to learn. In previous work, Scerri[25] has shown that a nary model, where agents discretize beliefs into three categories and communicate when their belief changes from one category to another can lead to fast cooperative convergence but has the potential for wild instability over time. In this paper, we build on that model, with agents being required to repeatedly choose roads in a network to get them from *home* to *work*. The time taken to traverse a road is a function of the number of other agents on the road when the agent begins to traverse the road. Randomization of the order of agent execution changes how long it will take a particular agent to traverse a road, because the order of arrival changes. This makes communication important for fast learning.

The first contribution of this work is to compare a human-inspire instance-based learning(IBL)[13] algorithm to the problem of multi-agent learning on the road congestion problem. The IBL model treats each trip from home to work as an instance for each road and weights instances based on several factors including recency when estimating time to traverse a road. For communication the IBL agents discretize their beliefs and communicate when their belief change discrete category, much like the navy model. We find that the agents using IBL do much better than the agents using the nary model and about the same as agents using a communication intensive averaging model. Interestingly, the IBL agents change routes more often than the other types of agents, partly due to the bigger differences in beliefs between the agents. We had several hypotheses about why IBL performed well which we evaluated by changing the behavior of the nary agents. The nary agents' behavior was closest to the IBL agents' when we restricted how often the nary agent could change behavior, suggesting that the IBL agents preference for reusing existing successful paths might be the key to its success. This is counter-intuitive since they actually changed more than the other agents.

The second contribution of this work is to look at system dynamics when the IBL agents and the nary agents are in the same system, learning in parallel to one another. We found that the overall system performance was improved by even a relatively small number of the IBL agents and that there were no negative effects on either type of agent. When there were only a few IBL agents in the system, they performed relatively better than the nary agents, but when

Appears in: *Proceedings of the Workshop on Multiagent Interaction Networks (MAIN 2013), held in conjunction with AAMAS 2013*, May 7, 2013, Saint Paul, Minnesota, USA.

there were many IBL agents all agents performed about the same. We also manipulated the social network on which the agents communicated and found that there was better performance when agents using the same learning algorithm were communicating with one another. This has interesting implications for systems where different types of agents are learning together, e.g., humans and agents in the same system.

Finally, the third contribution of this work is to look at how changes to the underlying system, in addition to changes due to learning, impact performance. One might expect that more numerical approaches will more quickly and effectively respond to change than a learner relying on experiences. However, we found no evidence of this. Instead IBL agents reacted very quickly and appropriately to underlying change, far better than the ternary agents. From these experiments we can see potential for using IBL in multi-agent learning settings and exploring these other settings is a key area of future focus.

2. MODEL

We use a model that appears in [25] and has been shown to produce interesting multi-agent learning dynamics. The model consists of agents A, places P and edges G over some number of iterations. Each agent $a \in A$ has some place, $p_{home} \in P$ where it starts each iteration and some place $p_{work} \in P$ where it must get to each iteration or day. To get to p_{work} it must use edges connecting places. Individual edges $g \in G$ connect exactly two places. The agents task is to get from p_{home} to p_{work} most quickly each iteration.

The time that it will take an agent to traverse an edge depends purely on the number of agents already on the edge when it gets to the edge. Specifically, the time taken by an agent is $10 + n^3_{already}$, where $n_{already}$ is the number of agents on the edge when the agent reaches it. The simulation randomizes the order the agents execute so that in one iteration an agent might be the first on the edge and have a very short travel time and another iteration it might be tenth onto the edge and have a very long travel time, even if none of the agents change their routes.

This model has two important features. First, the agents will get very different perspectives on speed of a edge, based on exactly when they get onto the edge. Hence, either many iterations or cooperation is needed to create an accurate model. Second, busy edges heavily penalize the agents, just a few extra agents on a edge will dramatically slow the last few agents down again making cooperation important.

For experimental purposes, there are only ten different p_{home} and p_{work} for 200 agents. This makes for more interesting traffic congestion problems, with more extreme cases, and requires more coordination among the agents, but, as was shown in [25] does not qualitatively change the system dynamics.

In every iteration, each agent uses a model of the graph to plan a path from p_{home} to p_{work}. The agents use a simple A* algorithm [24] to do the planning based on their current model of edge traversal times. Agents are risk neutral, trying to minimize expected travel time. They then execute their plan without adapting to observed conditions. At the end of an iteration, the agents can communicate about what they observed. The model the agent plans with and the information it communicates are described below.

It is assumed that each agent plans selfishly, but communicates truthfully and cooperatively. We are interested in two primary metrics. First, the average time it takes for an agent to get from p_{home} to p_{work}. Second, as the agents build their models and adapt their plans to the changing models, the average transit time will change. As a secondary measure, we are interested in the change in average transit time over time.

Communication Network.

The agents are organized into a social network where they can only communicate directly with a small subset of the rest of the agents. Information is propagated through the network in a peer-to-peer manner. Unless otherwise noted below, we use a random network with degree 5 to connect the agents.

2.1 Agent Reasoning

The agents have to choose a path that will most quickly get them to their destination, based on experiences so far and from experiences communicated from other agents. The optimal strategy might be one that considers likely plans by others and the changes they will make, given their previous experiences. However, this is typically infeasible. Cooperative agents with low cost communication might coordinate in advance to balance the routes, but for the purposes of this work we assume that to also be infeasible. Even if we assume that agents take other agents' plans into account, the game-theoretic Traveller's Dilemma [3] applies: If one agent A anticipates another agent B's reaction, A would also anticipate B's anticipation of A's reaction, and so on. Provided the game is played for a finite number of rounds, rational players will end up with a poor solution. Rational agents will be faced with a computation that does not scale. Moreover, if communication has any non negligible cost, any agent will only have partial information about traffic on edges over time.

We aim for any solution for this problem to deal with limited communication bandwidth, learn (quickly) to achieve acceptable performance, adapt to changing network dynamics. Human decision-making addresses fulfills these desiderata and is hence an inspiration for one of the approaches below. Below we describe three models for reasoning about the road network, the first having the agents only characterize a road as *slow*, *medium* or *fast*, the second is a human inspired instance-based learning approach. The third uses a simple moving average of expected times for each edge. Using any of these models the agents estimate the time taken to use a particular road and use A* planning to compute their expected fastest route, excluding any reasoning about how other agents might change their behavior. Notice that the agents are generally moving to a Nash Equilibrium, where, at least according to their local models, they have no incentive to change behavior. However, as has been noted before, even if the agents do reach a Nash Equilibrium, it may be the case that the outcome is far from the social optimal outcome[17, 14].

2.1.1 Ternary Model

In the ternary model, agents only track whether they believe a edge is *slow*, *medium* or *fast*. The agents keep a normalized frequency distribution of the observations for each of the edges, decayed over time. Specifically, for each edge e, the agent has model $M_e = \{p_{slow}, p_{medium}, p_{fast}\}, p_{slow} +$

$p_{medium} + p_{fast} = 1$. When an agent gets an observation of a particular category it adds β_{local} for a local observation and β for a communicated observation to the relevant p and then normalizes. For example, initially $M_e = p_{slow} = 0.33, p_{medium} = 0.33, p_{fast} = 0.33\}$, $\beta_{local} = 0.1$ and the agent observes an edge to be fast, $M' = \{p_{slow} = 0.302, p_{medium} = 0.302, p_{fast} = 0.395\}$.

The agents take the most probable category, $\max M$, and plan as if that was the case. In the experiments below, an edge in a particular category is assumed to take time 300, 156 and 12 for p_{slow}, p_{medium} and p_{fast}, respectively, corresponding to the average time when approximately 3, 7 and 11 agents also use the edge reasonable approximation of the typical expectations. When $\max M$ changes for an edge, i.e., when the agent's belief about an edge changes categories, it communicates the new category to its direct neighbors in the social network.

This model was designed to make communication easier. The agents communicate whenever their model changes from believing the edge falls into one category to believing the edge falls into another speed category. Agents receiving communications about category changes need to decide how to integrate the measurement into their model. A communication will occur based on a number of observations building up belief in some category, so it could be weighted more heavily than a local observation, which is a single data point. Previous work has shown this model to lead to very fast learning but unstable dynamics.

2.1.2 Cognitive Model

Cognitive agents implement a cognitively motivated aggregation mechanism that forms their beliefs. As in the ternary model, their communications are quantized and occur whenever their belief about a road changes. The same A* is used to plan paths. However, their estimates about the speed of each road are based on instance-based learning (IBL, [13]). IBL, related to memory-based learning, stores a datapoint (episode) with the speed of a road whenever it is traveled or when agents receive a communication. A speed estimate can then be derived as the average of all episodes associated with the road, weighted by the episode's *activation*. Activation is determined by a function that rewards experience (a large set of episodes), but discounts older information (decay). Activation has been shown to predict the availability of information in human memory in psychological experiments [2].

In detail, activation of an episode e consisting of a road speed (utility) and time, $< u_e, t_e >$ is given as

$$A_e = (t - t_e)^{-0.5}$$

t is the current time. The decay exponent is the default that is empirically realistic in human experiments. Our implementation uses an highly precise approximation of the above activation function that omits to store all but the n latest episodes. If a road is represented by a series of episodes R involving the road, then the expected speed of a road, $U(R)$ is derived as

$$U(R) = \frac{\sum_{e \in R} u_e e^{A_e/T}}{\sum e^{A_e/T}}$$

$T = 0.25$ is a parameter (*temperature*). If R is empty, we assume a default speed, U_β for the road. The agent's performance is sensitive to U_β, which represents a measure of pessimism (we do not optimize U_β and choose 0.0 as the most optimistic value).

Instance-based learning and the activation function have several desirable properties in our context. Activation increases during early iterations and allows the model to quickly differentiate between fast and slow roads. Activation is less affected by presentation of changes concerning frequently travelled roads.

2.1.3 Averaging Model

The Averaging Model computes a form of empirical ceiling: it is information-hungry, assuming that communication is free and unconstrained. It is the simplest model an agent can have of the graph is to store the average time taken by agents traversing that edge. Since the utilization of an edge will change over time, a moving average is used to keep the model updated with respect to the current situation.

Communication using the averaging model leads to problems of double counting. If agents simply share their current estimates with each other, where those estimates use both their own observations and communication from other agents there is a possibility that individual observations can end up being taken into account multiple times and skewing the averages. Various consensus algorithms that essentially ignore this effect have been developed and shown to work reasonably well[29, 20]. In this work, we take the conservative approach and require that agents share actual observations, which is expensive in terms of communication, but leads to principled, accurate averages. Hence, every time an agent traverses an edge, it communicates the time it took to traverse that edge to its direct neighbors in the social network.

The agents estimate for an edge is simply $e_i' = \alpha e_i + (1 - \alpha) obs$, where e_i is the current estimate for the edge and obs is the new observation for the edge, whether communicated or observed locally. In this paper, we use $\alpha = 0.95$.

3. EMPIRICAL EVALUATION

In this section, we empirically examine the three models on the congestion problem described in Section 2. The evaluation is split out into three parts, with each part aimed at looking in depth at one of the hypotheses introduced in Section 1. Unless otherwise stated, for each experiment below we use the following experimental parameters.

Table 1: Default experimental parameters.

Parameter	Value
Agents	200
Locations	100
Roads	Small worlds
Days	200
Runs	100

3.1 Instance-Based Multi-Agent Learning Dynamics

The key challenge for multi-agent learning is that all the agents are simultaneously learning, making the learning environment non-stationary. Learning from instances in a non-stationary environment is not an intuitively effective technique. However, humans, who arguably use a type of IBL,

are highly capable of learning in non-stationary environments. Our first experiments are aimed at looking at the performance of IBL on the congestion problem. Figure 1 compares the IBL, Ternary and Average models. Each model shows some improvement over time and some initial poor performance as the space is explored. The highly communication intensive and, for a human, computationally challenging Average model and the IBL model achieve about the same final level of performance and have about the same initially poor performance. Both do better than the Ternary model in the long run, although the Ternary model more quickly finds decent solutions.

Since the IBL and Average models end with about the same performance, it is tempting to conclude that they work in about the same way. However, Figures 2 and 3 show that they actually achieve the result with quite different dynamics. Figure 2 shows the average number of agents that change the path they take from the day before. The ternary model oscillates because beliefs take some time to change. More interestingly, IBL consistently changes more than Average. IBL agents change paths substantially more often, but the net result is the same as the Average agents. It is infeasible to determine exactly what is occurring, but it appears that IBL switch between approximately equal paths due to the noise in their relatively sparse data, while the Average agents have aggregated more data leading to more stable choices.

Figure 3 shows a snapshot of the variance in beliefs of the agents at the end of the 200 days. Specifically, for each road segment we computed the variance in the time the agents estimate it would take to traverse that road. These variances were then discretized and made into a histogram, with variances > 50 put in the 50 bin for clarity. The higher the variance the more the agents disagreed about how long it would take to traverse the road. Each of the three models lead to distinctly different patterns. The Ternary case often has all agents in agreement and never has large disagreements between agents due to the way beliefs cascade across the network and because the agents only allow a road estimate to have one of three values. The Average model shares more information leads to slightly lower variance overall than IBL, though the IBL has many more roads with very high variance, indicating complete disagreement. It is insightful to see that better performance was had when the agents had different models of the environment, many of which must actually be wrong. We can conclude that Average and IBL achieve approximately the same results, with very different algorithms and with distinctly different internal dynamics.

Conceptually, IBL does several things differently to Ternary. To try to understand what the cause of the different behavior was, we manipulated Ternary in several different ways. First, we artificially prevent Ternary agents from changing each step to mimic the IBL's preference for reusing previous paths. Second, we decay the learning rate so later data has less effect on Ternary, to mimic the way IBL instances aggregate. Finally, we change the default value for Ternary for unknown roads to match the default for IBL. Figure 5 shows that each of these changes improves Ternary performance, but preventing them from changing each step has the biggest effect. This is similar to other multi agent learning algorithms that improve performance by allowing only some agents to change at any time.

Figure 4 shows how communication networks influence the

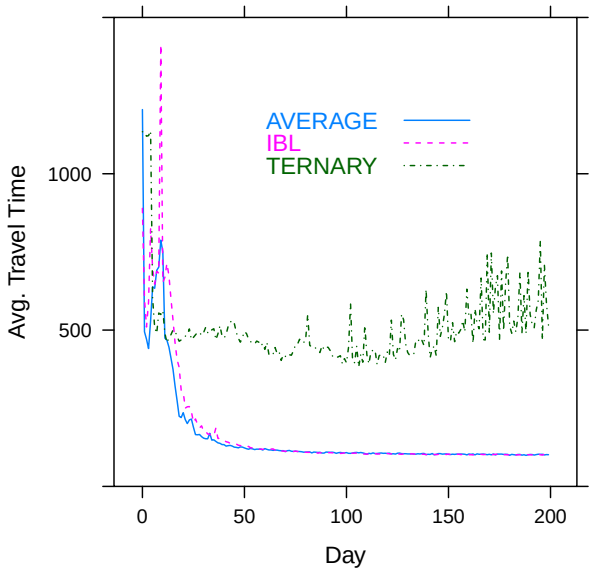


Figure 1: Comparison of Ternary, Cognitive and Average model agents over 200 iterations.

IBL agents. Curiously, blocking communication works similarly well as communication on fully connected and random network structures. These networks share information most evenly across the team, while ring and, to a lesser degree, small worlds networks compartmentalize information into neighborhoods. Although the effect is not very big, the data represents many simulation runs so the differences are real.

3.2 IBL and Ternary Models Interacting

IBL agents can be thought of as a simple model for how human learning might occur and Ternary agents can be thought of as a reasonable, low communication agent approach to cooperative learning. Future systems are likely to have humans and agents learning together and influencing each other. Hence, it is informative to look at what happens when IBL and Ternary agents are learning on the network at the same time. Figure 6 shows the joint average performance when there are different ratios of IBL and Ternary agents. Notice that it takes relatively few IBL agents to give the whole system an improvement in performance. Having different types of learners in the same system not only does not hurt performance, it actually helps the weaker learners do better.

Recall that the agents only communicate with their neighbors in a fixed social network. Figure 7 looks at what happens when we manipulate the network so that neighbors in that communication network are more likely to be of the same type. In the *Random* case the agents are randomly distributed across the social network so a neighbor is equally likely to be of either type. In the *Clustered* case, the overwhelming majority of the social network connections are to agents of the same type, but there are a small number of links, about 10, that connect the clusters so that information can move between agent types. Finally, in the *Separate* case there are no links from agents of one type to the other.

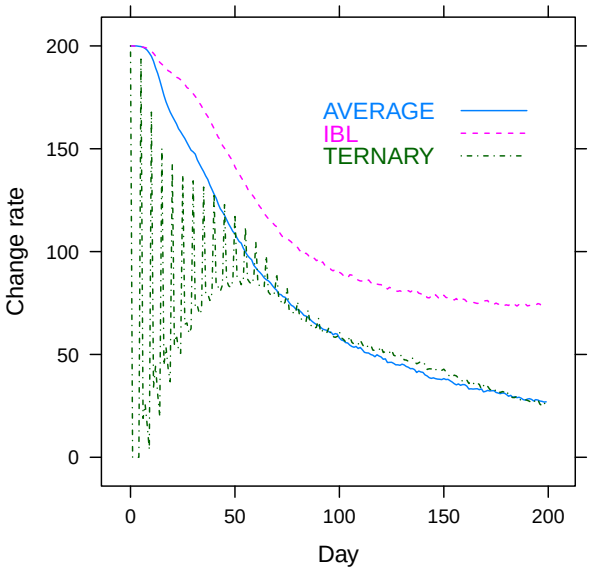


Figure 2: Rate of belief changes.

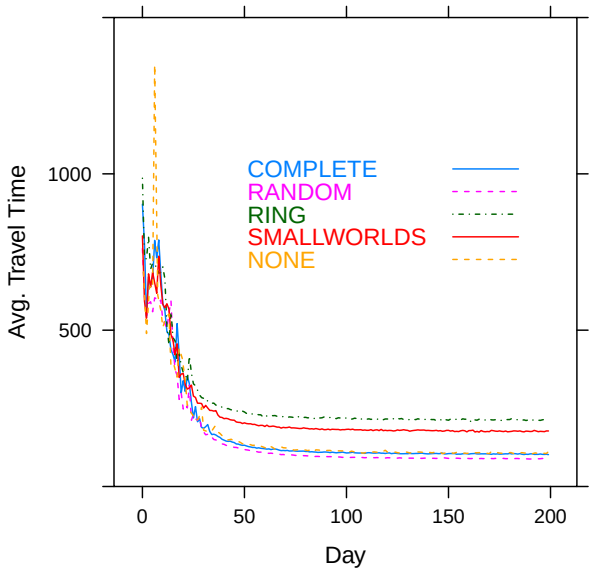


Figure 4: Average travel times for IBL model agents as the social network facilitating their communication is varied.

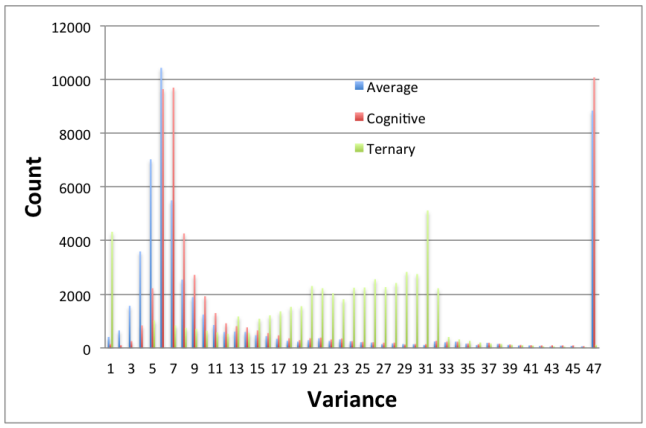


Figure 3: A histogram of the variance in estimates per road for each of the model types.

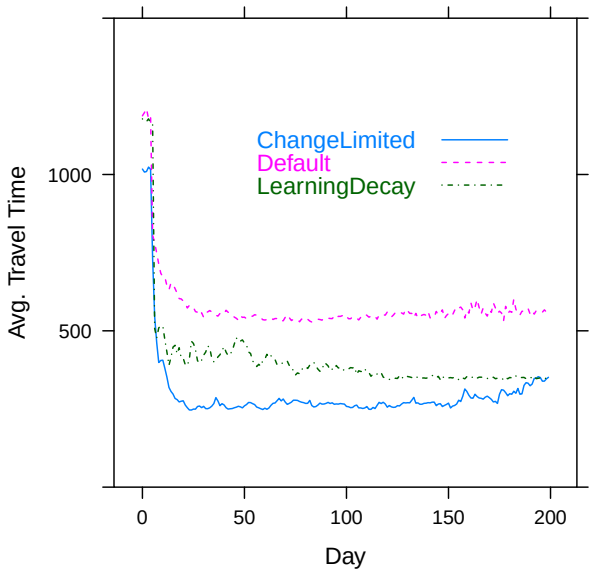


Figure 5: Results of Ternary manipulated in ways to make it work more like IBL.

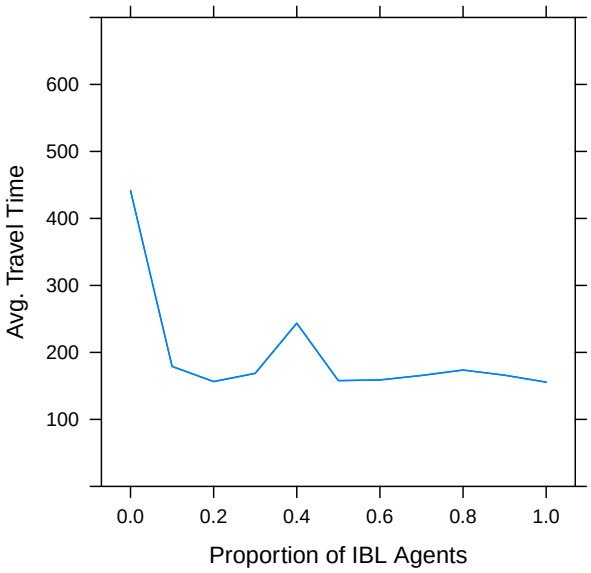


Figure 6: Different percentages of IBL and Ternary agents in the same system and the resulting average travel times.

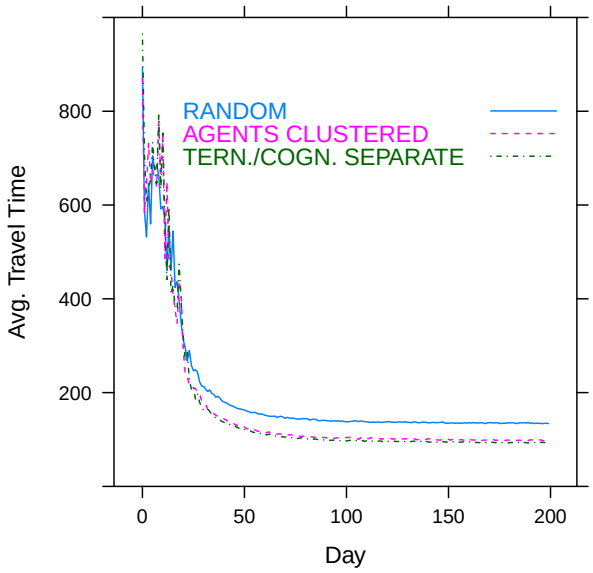
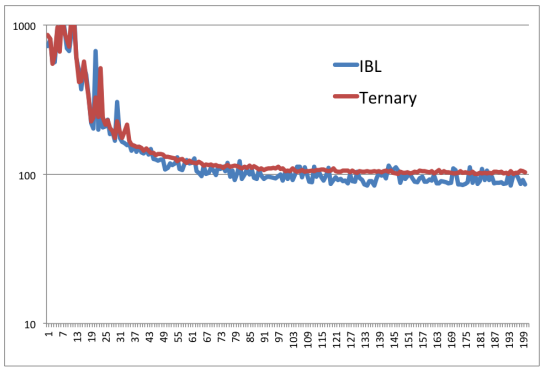


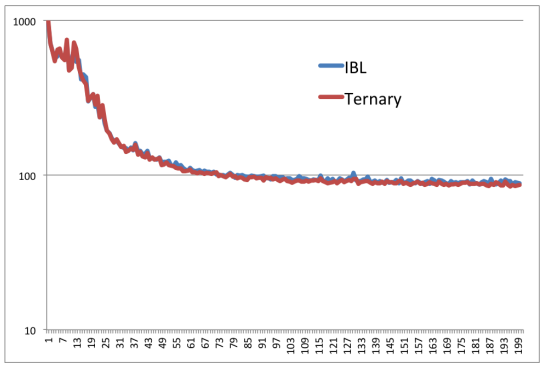
Figure 7: Different arrangements of IBL and Ternary agents within the same network.

Being completely separate is very slightly better than being mostly separate but either of these cases are better than mixing all the agents together. This suggests that the different types of information the agents are producing and the dynamics of the way they are doing it are more effective when shared with other agents of the same type. This has interesting implications for the design of future mixed human-agent systems.

In Figure 8 we show the relative performance of mixed Random graph networks of IBL and Ternary agents. Figure 8(a) shows the case with 20 IBL agents and 180 Ternary agents and Figure 8(b) shows 100 IBL agents and 100 Ternary agents. Notice that when there are only a few IBL agents they have a noticeable advantage over the Ternary agents, i.e., although they are using the same roads and are all interfering with each other, the IBL agents do relatively better. This advantage has disappeared when there are equal numbers of IBL and Ternary agents. The effect disappears smoothly as the number of IBL agents increases (not shown). If we think of IBL agents as being similar to humans and Ternary as being more like agents, this experiment hints that a small number of humans in an otherwise agent learning environment may do relatively better than the agents, although, as shown above, may improve the whole system's performance.



(a) 20 IBL, 180 Ternary



(b) 100 IBL, 100 Ternary

Figure 8: The relative performance (average travel times) of the different agent models in the same system. Lower values indicate higher performance.

3.3 Disruptions

Intuitively, learning from instances is likely to behave differently to learning moving numerical estimates when there are changes to the underlying system. Here we look at two different types of disruption to the underlying system, the addition of roads and the addition of agents, and the effects on the dynamics for each of the agent types. In the first case, one new road is randomly added every 20 days. The resulting dynamics are shown in Figure 9. The left most graph

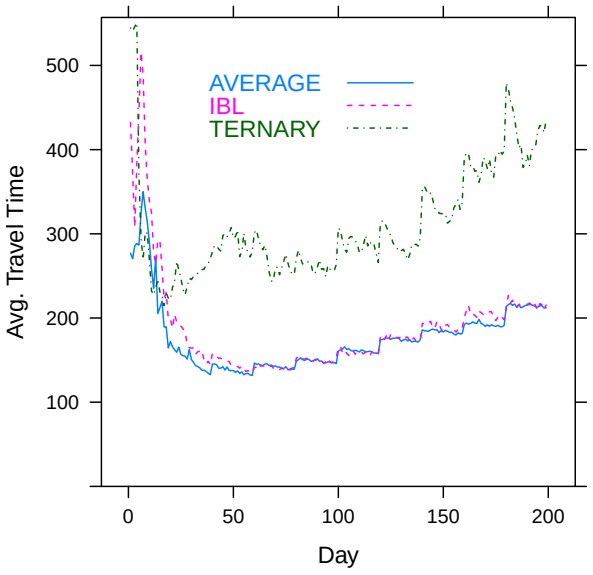


Figure 10: The impact of adding agents over time on average performance.

shows the average time over all 200 days, while the other two graphs show in detail the few time steps around the disruption. Both Average and IBL spike dramatically as they try to exploit the new road, but then go back to their original paths after finding it to be unhelpful – for most of them because they all tried it at once. The Ternary model does not get as badly impact because of the information sharing, but also takes longer to recover. Figure 10 shows the travel times as five new Ternary agents are added each 20 days, starting with 150 agents to make the result more comparable to other results. Both the Average and IBL agents jump when the new agents are added, but then smoothly improve performance. The Ternary agents are more dramatically effected by the change and do not adapt quickly. As the environment gets more congested and the original agents have built up more learning data, it is appears that IBL is more affected by the disruptions. This is unsurprising as its learning rate is effectively lower at this point.

4. RELATED WORK

Using agents to manage congestion in road networks has been addressed from a variety of perspectives[5]. Learning of traffic light patterns has been of particular interest[11, 18]. Bazzan has previously found that sharing information between traffic lights does not necessarily help performance[10]. More general management of congestion has also been looked at extensively[6, 4]. Multi-agent learning is an extensively studied problem [31, 9]. Most work focuses on how individual agents should learn in the context of the team, e.g., [15, 30]. Multi-agent versions of reinforcement learning has been a particularly popular approach[9, 28, 21]. Cognitive models have been combined to explain learning in team settings, primarily in a qualitative way ([26]). [23] used decay in a model implemented within a cognitive architecture to show that decayed memory improves agent performance in a foraging scenario with multiple, communicating agents. Instance-based learning within cognitive models has been shown to explain human behavior in a number of cognitive decision-making tasks ([19, 22, 12]).

5. CONCLUSIONS AND FUTURE WORK

This paper has looked at multi-agent learning dynamics and performance with a human inspired instance-based learning approach and two more quantitative agent-like models. The cognitive, IBL agents benefit from a relatively simple learning model, combining a preference for well-known roads and exploration of unseen roads. The agents can, with relatively limited communication volume, spread across the road network and efficiently use shared resources. What may be key to the cognitive agent's performance is limited sharing of knowledge: because agents do not have access to precise road utility estimates of their neighbors, and because they only receive updates when the neighbor's (quantized) beliefs change, they may arrive at heterogeneous conclusions about which roads are best. This leads them to spread out more, without sacrificing much individual performance. Under this scenario, agents do not need to misrepresent their knowledge states to their neighbors.

When combined in the same system, IBL agents and ternary agents actually helped each other rather than hurting performance. This is promising for future human-agent systems that will learn with distinctly different approaches. Our immediate future work generalize these results to better understand how humans mitigate negative multi-agent learning dynamics and how agents and humans will work together. One aspect of reasoning that we hope to incorporate soon is how agents model other agents and anticipate their changes. This is likely to have interesting effects in human-agent contexts where models will be inaccurate.

6. REFERENCES

[1] A. Ahmed, P. Varakantham, and S.F. Cheng. Uncertain congestion games with assorted human agent populations. 2012.

[2] John R. Anderson. *How can the human mind occur in the physical universe?* Oxford University Press, Oxford, UK, 2007.

[3] Kaushik Basu. The traveler's dilemma: Paradoxes of rationality in game theory. *The American Economic Review*, 84(2):pp. 391–395, 1994.

[4] A.L.C. Bazzan. *Multi-agent systems for traffic and transportation engineering.* Information Science Publishing, 2009.

[5] A.L.C. Bazzan. Opportunities for multiagent systems and multiagent reinforcement learning in traffic control. *Autonomous Agents and Multi-Agent Systems*, 18(3):342–375, 2009.

[6] N. Bhouri, S. Haciane, and F. Balbo. A multi-agent system to regulate urban traffic: Private vehicles and public transport. In *Intelligent Transportation Systems (ITSC), 2010 13th International IEEE Conference on*, pages 1575–1581. IEEE, 2010.

[7] M. Bowling and M. Veloso. Multiagent learning using a variable learning rate. *Artificial Intelligence*, 136(2):215–250, 2002.

[8] W. Burgard, M. Moors, D. Fox, R. Simmons, and S. Thrun. Collaborative multi-robot exploration. In *Robotics and Automation, 2000. Proceedings. ICRA'00. IEEE International Conference on*, volume 1, pages 476–481. IEEE, 2000.

[9] L. Busoniu, R. Babuska, and B. De Schutter. A comprehensive survey of multiagent reinforcement learning.

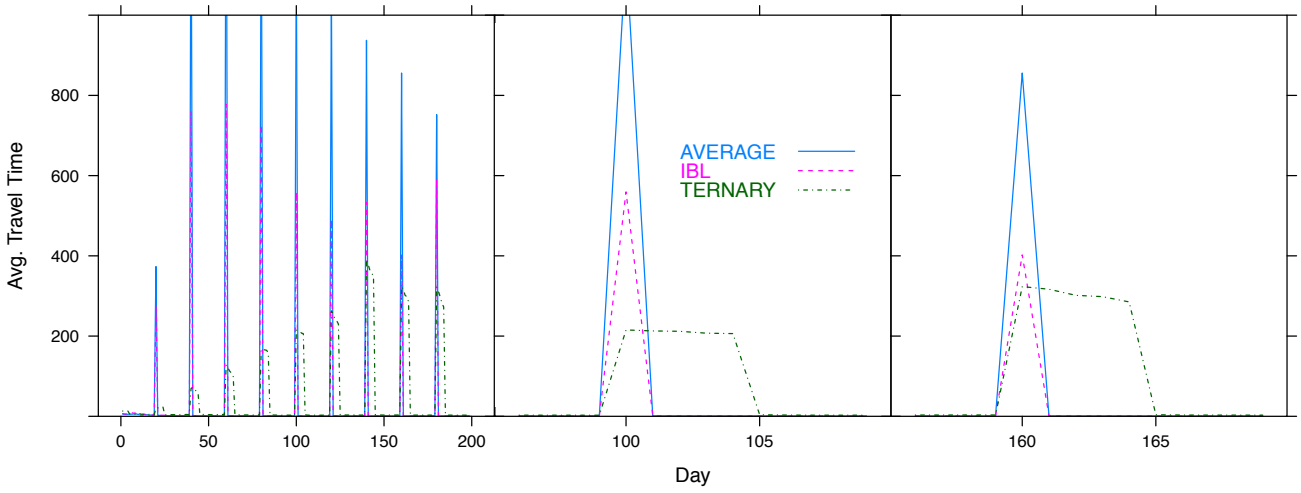


Figure 9: The impact of adding roads over time on average performance.

Systems, Man, and Cybernetics, Part C: Applications and Reviews, IEEE Transactions on, 38(2):156–172, 2008.

[10] D. De Oliveira and A.L.C. Bazzan. Multiagent learning on traffic lights control: effects of using shared information. *Multi-agent systems for traffic and transportation engineering*, 2009.

[11] S. El-Tantawy and B. Abdulhai. An agent-based learning towards decentralized and coordinated traffic signal control. In *Intelligent Transportation Systems (ITSC), 2010 13th International IEEE Conference on*, pages 665–670. IEEE, 2010.

[12] Ido Erev, Eyal Ert, Alvin E. Roth, Ernan Haruvy, Stefan M. Herzog, Robin Hau, Ralph Hertwig, Terrence Stewart, Robert West, and Christian Lebiere. A choice prediction competition: Choices from experience and from description. *Journal of Behavioral Decision Making*, 23(1):15–47, 2010.

[13] C. Gonzalez, F.J. Lerch, and C. Lebiere. Instance-based learning in dynamic decision making. *Cognitive Science*, 27(4):591–635, 2003.

[14] J.N. Hagstrom and R.A. Abrams. Characterizing braess's paradox for traffic networks. In *Intelligent Transportation Systems, 2001. Proceedings. 2001 IEEE*, pages 836–841. IEEE, 2001.

[15] M. Kaisers and K. Tuyls. Frequency adjusted multi-agent q-learning. In *Proceedings of the 9th International Conference on Autonomous Agents and Multiagent Systems: volume 1-Volume 1*, pages 309–316. International Foundation for Autonomous Agents and Multiagent Systems, 2010.

[16] S. Kalyanakrishnan, Y. Liu, and P. Stone. Half field offense in robocup soccer: A multiagent reinforcement learning case study. *RoboCup 2006: Robot Soccer World Cup X*, pages 72–85, 2007.

[17] Y.A. Korilis, A.A. Lazar, and A. Orda. Avoiding the braess paradox in non-cooperative networks. *Journal of Applied Probability*, 36(1):211–222, 1999.

[18] S. Lämmer and D. Helbing. Self-stabilizing decentralized signal control of realistic, saturated network traffic. Santa Fe Institute, 2010.

[19] Christian Lebiere, Dieter Wallach, and Robert L. West. A memory-based account of the prisoner's dilemma and other 2x2 games. In *Proceedings of the International Conference on Cognitive Modeling*, pages 185–193, 2000.

[20] R. Olfati-Saber, J.A. Fax, and R.M. Murray. Consensus and cooperation in networked multi-agent systems. *Proceedings of the IEEE*, 95(1):215–233, 2007.

[21] L. Panait and S. Luke. Cooperative multi-agent learning: The state of the art. *Autonomous Agents and Multi-Agent Systems*, 11(3):387–434, 2005.

[22] David Reitter. Metacognition and multiple strategies in a cognitive model of online control. *Journal of Artificial General Intelligence*, 2(2):20–37, 2010.

[23] David Reitter and Christian Lebiere. Social cognition: Memory decay and adaptive information filtering for robust information maintenance. In *Twenty-Sixth AAAI Conference on Artificial Intelligence (AAAI-12)*, 2012.

[24] S. Russell, P. Norvig, and A. Artificial Intelligence. A modern approach. *Artificial Intelligence. Prentice-Hall, Egnlewood Cliffs*, 1995.

[25] Paul Scerri. Modulating communication to improve multi agent learning convergence. In *Dynamics of Information Systems: Algorithmic Approaches*, to appear.

[26] Ron Sun, editor. *Cognition and Multi-Agent Interaction: From Cognitive Modeling to Social Simulation*. Cambridge University Press, 1 edition, May 2008.

[27] L. Tesfatsion and K.L. Judd. *Handbook of computational economics: agent-based computational economics*, volume 2. North Holland, 2006.

[28] M. Vasirani and S. Ossowski. A computational market for distributed control of urban road traffic systems. *Intelligent Transportation Systems, IEEE Transactions on*, (99):1–9, 2011.

[29] F. Xiao and L. Wang. Asynchronous consensus in continuous-time multi-agent systems with switching topology and time-varying delays. *Automatic Control, IEEE Transactions on*, 53(8):1804–1816, 2008.

[30] C. Zhang and V. Lesser. Multi-agent learning with policy prediction. In *Proceedings of the 24th National Conference on Artificial Intelligence (AAAIÕ10)*, 2010.

[31] C. Zhang, V. Lesser, and P. Shenoy. A multi-agent learning approach to online distributed resource allocation. In *IJCAI 2009, Proceedings of the Twenty-first International Joint Conference on Artificial Intelligence*, pages 361–366, 2009.