A Cognitive Hierarchy Model Applied to the Lemonade Game

Michael Wunder[1] **Michael Kaisers**[2] **Michael Littman**[1] **John Robert Yaros**[1]
[1]Department of Computer Science, Rutgers University
[2]Department of Knowledge Engineering, Maastricht University
{mwunder, mlittman, yaros} @cs.rutgers.edu, michael.kaisers@maastrichtuniversity.nl

Abstract

One of the challenges of multiagent decision making is that the behavior needed to maximize utility can depend on what other agents choose to do: sometimes there is no "right" answer in the absence of knowledge of how opponents will act. The Nash equilibrium is a sensible choice of behavior because it represents a mutual best response. But, even when there is a unique equilibrium, other players are under no obligation to take part in it. This observation has been forcefully illustrated in the behavioral economics community where repeated experiments have shown individuals playing Nash equilibria and performing badly as a result. In this paper, we show how to apply a tool from behavioral economics called the Cognitive Hierarchy (CH) to the design of agents in general sum games. We attack the recently introduced "Lemonade Game" and show how the results of an open competition are well explained by CH. We believe this game, and perhaps many other similar games, boils down to predicting how deeply other agents in the game will be reasoning. An agent that does not reason enough risks being exploited by its opponents, while an agent that reasons too much may not be able to interact productively with its opponents. We demonstrate these ideas by presenting empirical results using agents from the competition and idealizations arising from a CH analysis.

Introduction

System designers and researchers alike have become increasingly interested in the activities of interacting computer agents, especially in strategic settings (Niu et al. 2008). The problems of online auctions, botnets, and mobile networking are just some examples of areas where multiagent computing has arisen as a focal topic. In many cases, strategic reasoning is bounded because of constrained computation or uncertainty about other agents in the population. The usual techniques for these situations, such as traditional game theory and its recently developed cousin, algorithmic game theory, break down in cases of bounded rationality. In response, recursive models of players with varying reasoning capabilities have been proposed (Vidal and Durfee 1995; Gal 2006). Empirical competitions in simple games with complex dynamics, such as the game Rock-Paper-Scissors, have been well understood by differentiating levels of reasoning (Billings 2000; Egnor 2000).

In traditional game theory, analysts assume that players have access to unlimited reasoning ability or computational power, and are able to figure out a strategy that will prevent any disadvantage in expectation (Smith 1982). Algorithmic game theorists try to use greedy methods to find approximate solutions in the form of a system-wide equilibrium given agents with complete reasoning abilities. Games and competitions without these manageable properties are proliferating. One recent example is the lemonade game (Zinkevich 2009), which consists of three vendors attempting to maximize their selling space on a circular beach. A player's score only relies on the distance to opponents on either side, yet the iterated version yields complex interaction patterns.

In recent years, researchers have employed a behavioral model known as level-k thinking, or a cognitive hierarchy, to explain findings across a variety of experiments in the field of economics (Camerer, Ho, and Chong 2004). As the name implies, a cognitive hierarchy postulates the presence of levels, or steps, that naturally occur in human reasoning.

In this document, we propose a model for games that are heavily dependent on the types of agents likely to be encountered, with a special focus on the lemonade game (LG). If this model is substantially correct, then it means that the contest between individual LG strategies is usefully framed as a contest between levels of reasoning. While this framework certainly allows for various differences between agents classified at a certain level, it does specify that the levels present in a population can exert a bigger effect on the scores than particular implementation details. In this way, the model is like a map for LG strategy discovery and analysis.

While we do not arrive at a theoretical solution like an equilibrium for the lemonade game, we present a way to structure the likely reasoning of an unknown population. This model is based on the cognitive hierarchy approach, but we make some necessary adjustments to deal with multistep strategies and advanced reasoning. We advocate our method as a design principle for agents in similar games and situations. Just as importantly, the model aims to show what to avoid as the community continues to investigate multiagent learning practices in this domain.

The next section introduces repeated games, gives the ba-

Copyright © 2010, Association for the Advancement of Artificial Intelligence (www.aaai.org). All rights reserved.

sic concept of the Cognitive Hierarchy model (CH) using the example of the p-beauty game, and delineates the lemonade game rules and basic interaction patterns. The third section uses CH to derive the game's fundamental strategies, and finally experiments show the value of the CH process.

Background

Behavioral economics is a branch of economics concerned with the psychological biases and cognitive limitations of participants in strategic interactions. One particular area has focused on the so-called cognitive hierarchy that appears in populations playing certain types of games (Camerer 2003). A famous example of this model in action is in the beauty-contest game, in which the "judges" get the best reward by picking the most popular contestant in the pageant (Keynes 1936). In the game theory community, it could refer to any competition in which a player receives the most benefit from some position in response to an aggregate social decision. In the p-beauty contest version, m players submit a number x from 0 to 100 and the winner is the one who guesses closest to p times the average value, $p\sum_{i=1}^{m}\frac{x_i}{m}$, where $0 \leq p < 1$. The Nash equilibrium of a p-beauty contest is 0. However, researchers have demonstrated that groups do not initially play this result (Ho, Camerer, and Weigelt 1998).

A series of these findings (Costa-Gomes, Crawford, and Broseta 2001; Wang, Spezio, and Camerer 2009) prompted economists to propose that people are making decisions based on a model that combines the concept of base strategies with steps of thinking or reasoning. In this model, players first identify the most uninformed strategy. They then proceed to take steps of reasoning in response to that strategy until they have either lost the ability to go further or assume that not enough rivals will do so to justify the effort. While the bottom, or base, strategy is usually quite clear, the idea of a step of reasoning requires clarification. Here, a *step of reasoning* is a unit of problem solving, where the problem is to find the best possible action in response to the likeliest population that either is known or can be considered. Pure computational neuroeconomics might define a step of reasoning (SOR) as some step of computation made toward the goal of maximizing reward. In the p-beauty game, the base strategy is a uniform pick and the average of the base is 50. Each reasoning step is a simple multiplication of the average of the previous level by p. In repeated or stochastic games, a step can be a complicated optimization procedure, depending on the game and types of other agents. We would add a focus on new capabilities at each level that derive from more sophisticated views of the world. The next section will explain this feature of the model in more detail.

One reason this Cognitive Hierarchy model (CH) may map into artificial agent design is that programmers share the same psychological biases. A CH analysis provides a framework to explore the spectrum of potential opposing players. An accurate model for estimating the most likely strategies gives a head start to agents built upon that framework. This approach works best when the base strategies and the reasoning step are well defined.

Recently, the lemonade game was introduced to demonstrate the interaction complexity that can arise from simple rules (Zinkevich 2009). The game is played by three lemonade vendors on an island with n beaches, where typically $n = 12$, arranged like the numbers on a clock. Each morning, the vendors have to set up on one of the beaches, not knowing where the other vendors will show up on that day. Assuming the beach visitors are uniformly distributed and will buy their lemonade from the closest vendor, the payoff for the day is proportional to the distance to the neighboring lemonade vendors.

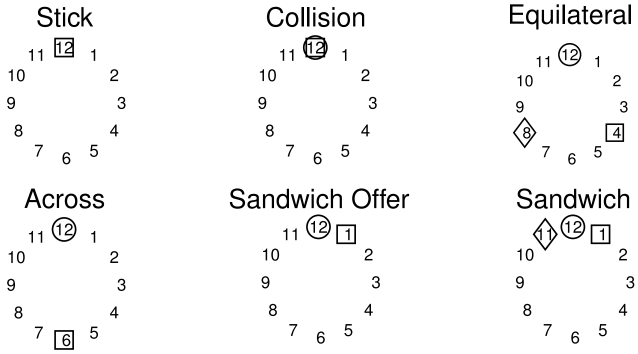

Figure 1: This figure depicts key strategic patterns of the lemonade game. Each of the six diagrams refers to a (partial) joint action, and similarly a strategic move by □, expecting opponents to play ◇ and ○. As the domain is on a ring, the patterns are rotation insensitive.

In game-theoretic terms, LG is a 12-action normal form game on a ring, where the payoff function equals the sum of distances to the right and left neighboring vendor. As a corollary, the cumulative payoff of the three players is 24. If two or three vendors share a spot they share the profit, 6 or 8 each respectively. The only exceptional formation is when two agents conflict by choosing the same action (Collision), they receive a reward of 6 and create the most favorable condition for the third agent who receives the maximum of 12. The game is played repeatedly for T days and the joint action is observable. T is set to 100 so that agents can learn about the opponents' behavior from previous rounds.

The dynamics of this game are particularly interesting because it involves a sense of competition, as the gains of one always have to be compensated by the loss of others, as well as a sense of cooperation, because two agents can coordinate a joint attack on the third. Figure 1 shows an overview of the key strategic patterns in the LG. Each agent has to choose an action, and the simplest move is to stick with the initial action from then on (Stick). The Equilateral pattern splits the payoff evenly into 8 for each agent, but from a risk perspective is dominated by the cooperative action Across. Once two agents coordinate on the action Across, it leaves the third agent with a payoff function of 6 for all actions. This action has been found to be the most frequent and stable cooperative arrangement in the competition, although a successful Sandwich gives a higher profit to the cooperating agents. In particular, Across can be played by □ with a non-responsive but predictable player ○, while the Sandwich as a joint action of □ and ◇ usually needs to be initiated by an Offer, and the second contestant needs to recognize and respond to the Offer. Note that □ actually risks a lower score by playing next to O. Furthermore, the victim can easily notice this attack and escape using another action. As a result, Sandwich is unstable despite delivering the near-optimal payoff 11 for each perpetrator and the minimum payoff 2 for the victim.

In contrast to the p-beauty game, which has a single parameter to optimize, LG allows for complicated policies that involve signaling and planning for cooperation. While the former is more common in the economics literature, games of the second class are usually investigated in computer science, where repeated games are used as a generalization of Markov decision processes to non-stationary environments including multiple agents.

Unlike the LG, the p-beauty game has a single Nash equilibrium where every player submits the number 0. Note that there is no unique equilibrium in the LG because a player will get the maximum score as long as he is on the side of the island with more space. However, as we have already mentioned, the stable Across pattern has the property that the players who set up in this way will be in equilibrium no matter what the third player does. Shifting a player one spot from Across has the same property but is slightly less efficient for those two opposing players.

Levels of Reasoning

The lemonade game is an ideal example of competitive collaboration. That is, a player able to convince another player to cooperate with it can achieve a higher average score to the disadvantage of the third player. Of course, each player has to choose the more friendly player to attempt to cooperate with, with the knowledge that those attempts will be tracked by the other players. Ultimately, the two players who work together best will achieve the highest scores.

While it appears that players have many repeated turns for observation and experimenting, in reality many games are settled in the first several rounds, as agents seek partners and mutual history is established. Again, cooperation, however it is defined, is self-reinforcing. Therefore, this game puts a premium on speed over depth when finding optimal actions. The advantage of fast movers argues in favor of heuristics over learned strategies. In addition, if a player has an idea about which other strategies to expect in the population, it will provide additional benefit.

In the LG, the defining characteristic of level k does not exist at levels $0...k-1$, and emerges as a direct response to the vulnerabilities of level $k-1$. Due to the three-player structure of the LG, there are two ways to determine how a new level arises from the previous one. In one case, a single player at level k is looking to optimize against two players at level $k-1$. In the other, two level k strategies try to cooperate against one player at level $k-1$. There is a zero-sum game played between a player and the other two agents so both problems are relevant for analysis. We will define a step of reasoning (SOR) in the LG to be a policy that maximizes the average of the first case first, and the second next if there are several possible strategies available.

Table 1: Reasoning levels

Level	Meta-strategy
L0	Stick with probability x, random action otherwise
L1	Play at position Across from most consistent player
L2	Stick unless losing, may contain elements of L1
L3	Sandwich with L3 vs. Stick, plus elements of L2

As with an inductive proof, the reasoning level 0 (L0) forms the basis for iterated strategies of the rest of the hierarchy (see Table 1 for examples of levels). First, these base strategies need to be defined, and subsequently the higher layers can be constructed by iteratively applying the reasoning step. In many games, a base strategy of a single uniform distribution over all actions suffices. In repeated games like this one, there exists another trivial action Stick, which leads to the basic notion of stickiness in a strategy. Stickiness, as measured by the likelihood that a player will remain in place, plays an important role in this game because it makes move prediction very simple. As such, it deserves a place among base strategies.

The general base level L0, therefore, is composed of uniform action and sticky action, with a parameter x to control the relative importance of each. The L0 strategies are defined by an initial random action and the probability x to Stick with the previous action in the following turn, or otherwise pick a new random action. For $x = 0$ or $x = 1$ L0 takes the form of a uniformly random (L0-U) or constant strategy (L0-C) respectively. Because this value x is unknown to opponents, it must be estimated over several time periods. Define $\hat{x}$ as the current estimate of x for an opponent. We will not require x to be fixed, but in keeping with the tendencies of LG and empirical observations it is expected (though not required for our analysis) to rise over time.

We propose the three additional levels in Table 1, which arise from calculating iterated best responses in LG. The levels function as the main building blocks, or meta-strategies, available at the start of a session of repeated LG. Each successive level responds to the levels below it. As a result, each lower level is open to an attack by some combination of higher strategies. We can also define these levels as new capabilities opening up in response to new challenges. We might consider that L1 views a world where opponents do not respond to others' actions and acts accordingly. L2 anticipates that others react quickly, while L3 recognizes the potential of advanced coordination.

Define x_i as the value of x for player i. $V(6)$ is the value of playing action 6 or at location 6. $V(Across(i))$ is the value of playing Across player i. The optimal strategy for player C when faced with two L0 players, A and B, is to play Across from the player with the higher x if that value of x is close to 1. In other words, we want to show how the level 1 strategy arises when faced with a reasonable degree of certainty that one of the players will Stick at its prior location. We will assume a high x for at least one player so that the best response is clear.

Consider $x_A > x_B$ without loss of generality, with A playing action 0. If $x_A = 1$, then C's payoffs for action n

and uniform random action of B are:

$$\begin{aligned}
V(n) &= E[score|Location(A) = 0] \\
V(n) &= \frac{1}{12}(6) + \frac{1}{12}(12) + \frac{max(0, n-1)}{12}(12 - \frac{n}{2}) + \\
&\quad \frac{11-n}{12}(6 + \frac{n}{2})
\end{aligned}$$

The first term is the event that B lands on C. The second term is the event that B lands on A. The third term is the event that B lands in the short distance between A and C, and the fourth term represents when B lands on the large distance side.

$$\begin{aligned}
V(0) &= \frac{11}{12}(6) + \frac{1}{12}(8) \\
&= 6.17 \\
V(1) &= \frac{1}{12}(6) + \frac{1}{12}(12) + 0 + \frac{10}{12}(6 + \frac{1}{2}) \\
&= 6.92 \\
V(2) &= \frac{1}{12}(6) + \frac{1}{12}(12) + \frac{1}{12}(12 - \frac{2}{2}) + \frac{9}{12}(6 + \frac{2}{2}) \\
&= 7.67 \\
V(3) &= \frac{1}{12}(6) + \frac{1}{12}(12) + \frac{2}{12}(12 - \frac{3}{2}) + \frac{8}{12}(6 + \frac{3}{2}) \\
&= 8.25 \\
V(4) &= \frac{1}{12}(6) + \frac{1}{12}(12) + \frac{3}{12}(12 - \frac{4}{2}) + \frac{7}{12}(6 + \frac{4}{2}) \\
&= 8.67 \\
V(5) &= \frac{1}{12}(6) + \frac{1}{12}(12) + \frac{4}{12}(12 - \frac{5}{2}) + \frac{6}{12}(6 + \frac{5}{2}) \\
&= 8.92 \\
V(6) &= \frac{1}{12}(6) + \frac{1}{12}(12) + \frac{5}{12}(12 - \frac{6}{2}) + \frac{5}{12}(6 + \frac{6}{2}) \\
&= 9
\end{aligned}$$

Therefore, the move Across from a player who remains in place will average 9 when the other player is random. Since this score is the best to be done against a uniform player, it is an optimal strategy to set up Across from a player if there is a reasonable confidence that it will not move away.

If $x_A < 1$, the strategy Across(A) may still be optimal in relation to a low x_B. The next actions depend on the initial locations of A and B. Imagine A plays 0 and B plays 11. Then Across(A) is optimal either if both A and B Stick or both are random next turn. The true comparison lies when only one Sticks.

$$\begin{aligned}
V(6) &= 9x_A(1 - x_B) + 8.92x_B(1 - x_A) \\
V(5) &= 8.92x_A(1 - x_B) + 9x_B(1 - x_A) \\
V(4) &= 8.67x_A(1 - x_B) + 8.92x_B(1 - x_A) \\
&\ldots
\end{aligned}$$

The other actions have worse payoffs than V(4) and are therefore suboptimal. We know that $x_A(1 - x_B) > x_B(1 - x_A)$, so V(6) is optimal.

Imagine A at 0 and B at 10.

$$\begin{aligned}
V(6) &= 9x_A(1 - x_B) + 8.67x_B(1 - x_A) \\
V(5) &= 8.92x_A(1 - x_B) + 8.92x_B(1 - x_A) \\
V(6) &> V(5) \\
0.08x_A(1 - x_B) &> 0.25x_B(1 - x_A) \\
x_A(1 - x_B) &> 3.125x_B(1 - x_A) \\
\frac{x_A}{1 - x_A} &> 3.125\frac{x_B}{1 - x_B}
\end{aligned}$$

If $\frac{\hat{x_A}}{1-\hat{x_A}} > 3.125\frac{\hat{x_B}}{1-\hat{x_B}}$, V(6) is optimal. We can continue in this manner for all initial positions. This strategy in fact requires a degree of sophistication, in that C must estimate the current values of $\hat{x_A}$ and $\hat{x_B}$. Across(A) works as long as x_A is much higher than x_B, or if recent observations provide evidence that $\hat{x_A} \approx 1$, discounting the weight of the early rounds.

L1 is constructed to maximize reward in the world of L0s that it perceives. Although no player in the L0 population will respond to the actions of L1, L1 can still prosper if it identifies the better player to play Across. Notice that this strategy can apply to a range of situations, such as when the other players are reasoning. This L1 strategy succeeds as well in the case where two L1 agents meet an L0 strategy with $x < 1$, as long as the L1s end up Across from each other. In fact, this case cements the above strategy because if C is faced with a L1 A player, C will wish to appear a better partner to prevent A from exploring B as a partner. This bit of reasoning adds to the sophistication of the L1 agent, but it also points to the elegance and symmetry of Across as a mutually beneficial strategy. While two opposing L1 players will do well against a uniform or semi-random player, the strength of L1 comes from the ability to quickly identify partners regardless of reasoning ability.

However, the narrow focus of L1 does create a weakness. Simply put, a single constant player (L0-C) has the advantage against two L1s because the L1s must move to implement their strategy. It is likely that at least one of them will jump across from L0-C. At that point, the other L1 is stuck, as it cannot take any action to get itself more than 6. In reality, the most likely situation is that both players Collide across from L0-C. The proof is that at each timestep, both players choose a partner to go Across. If one L1 chooses the other, then there is some chance that the other L1 moved also, and therefore both have to choose again. For every step the players are not yet partners, the likelihood of ending up across from L0-C increases. This weakness of L1 creates an opening for L2.

Since stickiness is prized by the L1 strategy, L2 pursues the strategy of playing Stick for as long as possible, even if no player is Across. If L2 can stay still long enough for two L1s to both select it as a partner, it will derive enormous benefit as the partner of both. While that outcome would be the best case scenario, L2 cannot ignore L0 either, and so should employ some moving ability. L2 therefore will remain constant unless there is some reason that it should not, such as getting stuck next to another constant or some other losing situation. The threshold for when to move, as well as the action to take, are design decisions that can be made any number of ways. Similar logic confirms this strategy for the case of two L2s versus one L1. If an L2 waits for some L1 to move Across from it first, it will wish to remain as stable if there is another L2, also. There is not much cooperation between L2s that can occur in this case.

The L2 strategy also contains a weakness. If it remains fixed for too long, it can suffer enormous losses in the event that the other two players can manage to team up against it by pinching the L2 in a Sandwich attack. We will consider L3 to be a player that is able to engage in this attack against fixed players with another L3. While an L2 could jump out of the middle of a Sandwich, the L3s might then consider the Sandwich attempt an excellent bonding experience and therefore use it to guarantee partnering with each other. In that case, the L3s still benefit at the expense of the L2. In addition, the constant strategy by definition cannot move and is therefore the perfect target for this kind of attack. In essence, the strength of L2, its consistency, is used against it by a pair of L3s. If the Sandwich attack works, the L3s will avoid the fate of two L1s against L0-C.

This strategy has an expectation of 11 for as long as the L2 (or L0-C) victim does not move and then 9 for the rest of the game, against a non-static L2. As such it is the highest that can be achieved against an L2 player. With an L2 opponent and equal chance against L2 and L3 for the other, it is best to play the Sandwich Offer against an L2. A single L3 against two L2s has the advantage of first motion, so even if its Sandwich fails, it can fall back into L1 mode with a partner of its choice. Therefore the Sandwich passes this test as well.

Sandwich is revealed as a cooperative move in a subgame composed of two players, A and B, and a third fixed player C. The pure defensive, and least risky, move for A is to go across from C. Thus, the Sandwich against C is analogous to a cooperative game for A and B, but it requires both players to notice that fact. The main feature of L3, in addition to the Sandwich move, is its tolerance of risk for the sake of deeper coordination. While L3 performs well against two L2s, a Sandwich offer can be marked by a shortsighted L1 agent as a noncooperative move if it is not aware of this subgame. The main weakness of the L3 characteristic lies with the risk that the players B and C are seeking a reliable Across partner instead of a Sandwich partner. Only one submission of the LG competition reached this level of reasoning, hence further levels are omitted in the analysis. As a rule, a well-implemented level-k strategy can perform moderately well against previous levels, even if it is not optimized against all possible populations with any particular average level of reasoning.

Experimental Setup and Results

Up to this point, the levels in LG are just thought experiments. In fact, the basic elements of the CH account arose in a group of agents developed independently. This section shows the viability of the CH analysis by applying it to the open LG competition of Jan. 2010.

Tournament

The final competition consisted of a round-robin tournament of the lemonade game and the average score is taken for the purpose of ranking. The contestants and their scores are listed in Table 2.

Analysis of Strategies

Each contestant is evaluated over many games against a population of idealized agents to determine its characteristic reasoning level. The results are further supported by correlating the tournament outcomes of the competitors and their CH idealizations.

To apply the model to real agents, we must first classify each strategy by level. If a CH model is a good fit for LG, populations consisting of agents that correspond to a similar mix of levels should behave in roughly the same way as their idealized counterparts. Since each level has its unique strengths and weaknesses, performance depends on the makeup of the population and specifically the relative frequency of each level. For the purposes of this paper, we classify a strategy by inspecting how it scores against idealized strategies from each of the levels we identified.

Consider τ to be the average reasoning level L in the population. One way to highlight crucial properties of a given agent is to play it against populations composed of each level, or weighted mixes of adjacent levels. We found the clearest classification method by setting the level frequencies of a τ-population by linear interpolation of the integer levels immediately higher and lower. So, a population with a τ of 1.2 would be made of 80% L1s and 20% L2s. [1] That is, each of the two rival agents in each trial would be L1 with an 80% probability, and L2 with 20% probability. Any method that provides good separation would be adequate to the task of sorting individual strategies. The exact proportion of the levels at each point on the line does not matter as much as the ability to pinpoint how an agent does against two adjacent levels at once. While it might also make sense to inquire about how a player performs against non-adjacent levels, like a mix of L0 and L3, it is not necessary and more difficult to interpret.

As we vary τ according to this method, the population ratio shifts and so do individual agent scores against that population. The expected performance for a strategy at level k as τ rises would be to peak at some $\tau < k$ and then decline at a rate depending on the strategy's compatibility with others like it. Figures 2 to 4 demonstrate this trend for two sets of agents, the actual official competition conducted with nine participants in January 2010 and idealizations of these contestants. See Tables 2 and 3 for the agent descriptions.

Using the above criteria, we can classify each strategy with an estimated level (EL), while remembering that some agents do not explicitly execute any of the level-based strategies, and others incorporate some mix between them. Be-

[1] A more common method for estimating level-k frequency $f(k)$ in CH analysis uses the Poisson distribution, such that $f(k) = \frac{e^{-\tau}\tau^k}{k!}$. This method does not highlight the diverse strategies of LG and so is less convenient than interpolation, which has similar qualities.

Table 2: A list of official tournament submission strategies including their tournament average payoff per day (Score). An arrow indicates that the agent starts at one level, but sometimes transitions to a different one.

Name (Team)	Score	Est. Level	Description
EASquared (Southampton)	8.62	EL2 → EL1	Attempt to get Across from sticky opponents or train followy opponents to play Across from it.
ModifiedConstant (Pujara)	8.52	EL2	Randomly pick an initial location then Stick. If a number of turns with low utility occur, randomly pick a new location.
CoOpp (RL3)	8.51	EL1 → EL3	Continuously attempt to engage in Across. Attempt Sandwich in some situations.
MyStrategy (Waugh, CMU)	8.20	EL0 → EL1	Recursively compute best responses assuming everyone else plays best responses.
ACTR (Lebiere, CMU)	8.15	EL1 ↔ EL2	Cycle 3 strategies: 1) Stick, 2) Across weakest opponent, 3) Across strongest opponent.
GreedyExpectedLaplace (Schapire)	7.75	EL0 → EL1	Using Laplace Smoothing, predict each agents location and pick best reply given these opponents.
BrownLemonade (Brown)	7.67	EL0 → EL1	Use an ensemble of predictors to guess opponents' next moves and pick location with optimal utility.
FrozenPontiac (Michigan)	7.58	EL0 → EL1	Compute probabilities that agents play Stick, ERMS, rational (furthest from opponents) or n2p (next to the other opponent) and find the optimal location given these probabilities.
GregStrategy (Kuhlmann)	6.99	EL0	Move to a uniformly random location every turn.

Table 3: A list of idealized testing strategies.

Name	Level	Description
QLearner	L0→L1	Use simple Q-learning to choose Stick or Across opponent one or two.
OppositeCommonAction	L1	Pick the location Across from the historically most frequently used location.
StickyOpposite	L1	Pick a random partner to play Across and Stick until they move.
OppositeTheLeader Constant	L2	Stick until a threshold of loss is experienced, then move Across the current point leader (unless it is us).
SeclusiveConstant	L2	Stick until a threshold of loss is experienced, then move to a location furthest from opponents.
PinchTheConstantOrElse	L3	If a fixed player is identified, aggressively attempt Sandwich offer. If the attempt fails, punish the non-fixed player by playing Across from the fixed one.

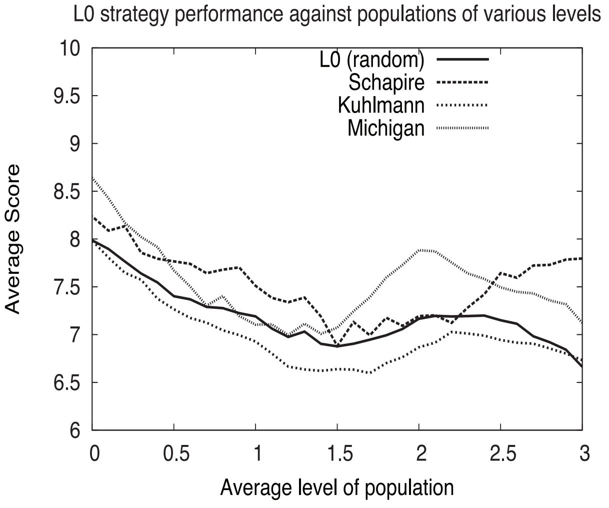

Figure 2: This graph shows how L0 (uniform or semi-uniform action) agents perform against various level combinations. From an average score around 8 against other L0s, these players drop against the higher, or faster, reasoners.

cause of the ambiguity of classification, and the variety of strategies allowed at each level, there are many possible population configurations for evaluation purposes. We follow a two-stage process for classifying agents by level. First, we build a sample population of representative agents for each level, and show how the level-by-level performance varies in a way predicted by the cognitive hierarchy model. Next, we play the competition agents against this population, and approximate their level by minimizing the sum of squared error of each agent's performance from the average performance of some entire idealized level. With these estimates it is possible to run a new competition to check if idealized players perform like the actual competitors. We used an iterative process like this one to design our own challenger for the actual competition.

It is possible that a more complicated agent may sometimes switch between the behaviors predicted at separate levels. That possibility does not undermine the fact that its overall reward is explained by its overall behavior, which derives from its mixture of levels. Since this behavior depends on the population an agent faces, it is meaningful to ask about the "levelhood" of its distribution of play. So an agent may perform like one level against one population, but very different from that level against another population. If

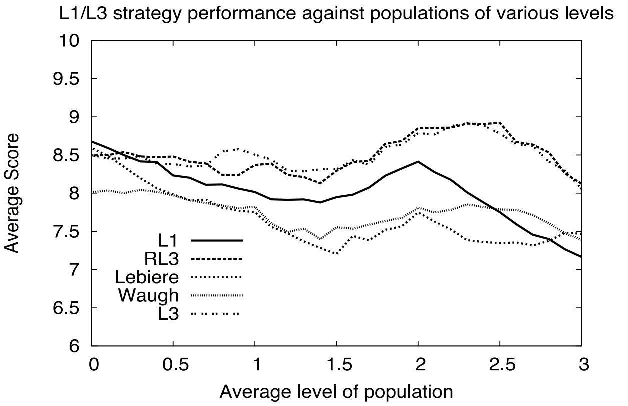

Figure 3: This graph shows the profiles of L1 and L3 agents. The L1 strategy starts off well but declines against other L1s, and then does well against mostly L2 populations. The L3 profile, while similar to L1, does much better against constant-based players and other L3s, as the model predicts.

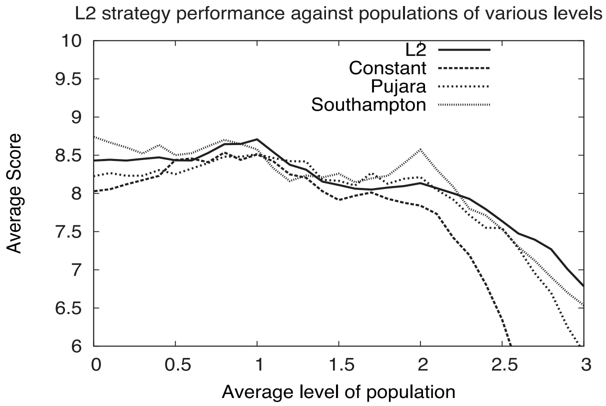

Figure 4: This graph shows how L2 agents perform against various populations drawn from across levels. These agents do best against a mainly L1 population, but decline thereafter, and especially facing L3s.

it fits another idealized level more consistently over all populations, it is considered to be more in line with that level.

The graphs demonstrate the performance of several level-specific strategies. The performance of each idealization is different enough to provide good contrast. Figure 2 shows how the L0 idealization and three below-average contestant strategies perform at a population averagin g L0 with a score around eight and drop rapidly against higher population levels. The higher level players also show the expected results. In Figure 3, the performance of L1 starts relatively high and drops as L1s become more common, then peaks again at a population of all L2. The L2 players of Figure 4 rise to a peak against an average reasoning level of L1, then drop swiftly. The L0-C (Constant), is included with the L2s because it shows a similar pattern. Notice that, for higher levels including mostly L3 players, L2 does not suffer as badly as L0-C. Finally, the L3 players roughly track the L1s, and furthermore do very well against L0-C or the mostly constant L2 players. These empirical results partially confirm that the level definitions were broadly correct, as the peaks and valleys occur right on target where the model predicts.

Our generic method classifies agents submitted to the LG competition with varying accuracy. Second place Pujara's agent behaves as a textbook EL2 (Figure 4). Several contestants score near the EL0 range (Figure 2). While the rest are harder to classify, the model still works as it should.

For example, the third place player RL3's agent tracks very close to EL3 (Figure 3). Its mix of level-based strategies causes it to begin with a chance of Stick, but then quickly switch to find a good opposite-based partner. Once it is Across from another participant, RL3 keeps track of how committed the partner is to remaining Across from it. In the event that RL3 ends up Colliding in the space Across from a constant player, it becomes open to Sandwich. While RL3 only intiates a Sandwich offer a quarter of the time, it is enough to qualify partially as an EL3 player, although its default strategy is EL1.

The winning Southampton agent (Figure 4) starts as a constant, and then initiates an L1-based strategy. Therefore, its performance and level behavior depend on whether it faces patient EL1s like itself. This version of delayed EL1 is meant to wait out impatient L1s, but still receive the benefit of EL1 action against everyone else. Against pure L1 idealizations, this strategy appears to score like an EL2. However, several clues give away its latent EL1 tendency, especially its strong performance against L0s and L2s, and a peak closer to $\tau = 0.8$ instead of $\tau = 1$ for well-fitting EL2s like Pujara. Figure 4 makes it clear why Southampton did well against this group of competitors, but Pujara would do better against an all L1 population. In the same way, RL3 would do better against a mix of L2s and L3s.

Finally, the remaining agents fit the scoring performance of either EL0 or EL1 (Figure 2). The Waugh entry attempts an advanced hierarchy-based best response algorithm. While it was the only agent to explicitly attempt to measure the level of others, the shortage of good data early on leaves this approach at a disadvantage. Several prediction-based learners apply a variety of methods to classify opponents, and thus try to outwit them. Unfortunately, by the time the learners find an optimal response, it is too late because the faster players have already matched up with each other, leaving out the slow learner. The end result is that while it is clear these agents are non-random, the observed performances roughly track those of an almost random player, EL0.

Figure 5 shows a mock tournament between the idealizations corresponding to the submitted competitors. These simplistic agents replace the submissions according to estimated level. As the outcomes of the two tournaments show, there is a close correlation between the scores of the submitted players and their level-based substitutes — a Pearson's r of 0.975. In fact, the ordering is almost identical, with a Spearman's rank correlation of 0.95. Hence, the Cognitive Hierarchy model provides a useful tool for understanding and predicting the performance of strategies in the LG.

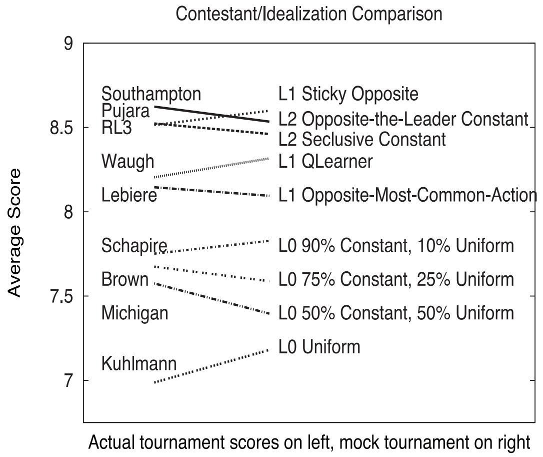

Figure 5: Comparison between the January 2010 competition contestants and corresponding idealizations. Each line segment connects the data point from the actual competition with the simulated result.

Conclusion

This article has introduced a Cognitive Hierarchy analysis for repeated games and applied it to the Lemonade Game competition. The high correlation between the mock tournament of representatives from the CH levels and the actual competition shows that the essence of the competitors has been captured by their CH idealization. In the competition, simple heuristics outperformed intricate learning schemes, suggesting that CH analysis might be preferable to domain-general best responses in strategic interactions. LG requires strategies to trade off speed and the dependence of reasoning on data, or depth. Those participants who opt for too much depth over rapid responses suffer against more structured strategies. Figure 5 is not meant to show the futility of learning in LG, but rather that players must employ some basic heuristics in the early stages of a game. If they do not, they risk getting classified as the less responsive, consistent, or cooperative partner. Despite the difficulty of behavior forecasting, there is no question that learning can play a role, especially among higher level strategies. However, that learning needs to take place in the proper space, or else a strategy will not have the capacity to react to basic heuristics. The top three players did adapt somewhat in response to their opponents. They did so by recognizing that they were not playing against distributions like those found in single-agent domains, but other players who understood the rules and were prepared to leverage them against slower players.

In sum, the CH analysis achieves good predictions of the strategies' performances. Furthermore, it has revealed characteristic properties of the LG. Future work will aim to show its applicability to further domains and establish the method as a framework to understand multiagent games of this kind.

Acknowledgements

The authors would like to thank the National Science Foundation for support on this project via NSF HSD-0624191.

References

Billings, D. 2000. The first international roshambo programming competition. *ICGA Journal* 23:42–50.

Camerer, C. F.; Ho, T.-H.; and Chong, J.-K. 2004. A cognitive hierarchy model of games. *Quarterly Journal of Economics* 119:861–898.

Camerer, C. F. 2003. *Behavioral Game Theory: Experiments in Strategic Interaction*. Princeton University Press.

Costa-Gomes, M.; Crawford, V.; and Broseta, B. 2001. Cognition and behavior in normal-form games: An experimental study. *Econometrica* 69(5):1193–1235.

Egnor, D. 2000. Iocaine powder. *ICGA Journal* 23:33–35.

Gal, Y. 2006. *Reasoning about Rationality and Beliefs*. Ph.D. Dissertation, Harvard University.

Ho, T.-H.; Camerer, C. F.; and Weigelt, K. 1998. Iterated dominance and iterated best response in experimental p-beauty contests. *American Economic Review* 88:947–969.

Keynes, J. M. 1936. *The General Theory of Employment, Interest, and Money*.

Niu, J.; Cai, K.; McBurney, P.; and Parsons, S. 2008. An analysis of entries in the first TAC market design competition. In *Proceedings of the IEEE/WIC/ACM International Conference on Intelligent Agent Technology*.

Smith, J. M. 1982. *Evolution and the Theory of Games*. Cambridge.

Vidal, J. M., and Durfee, E. H. 1995. Recursive agent modeling using limited rationality. *Proceedings of the First International Conference on Multi-Agent Systems (ICMAS)* 376–383.

Wang, J. T.-y.; Spezio, M.; and Camerer, C. F. 2009. Pinocchio's pupil: Using eyetracking and pupil dilation to understand truth-telling and deception in sender-receiver games. *American Economic Review*.

Zinkevich, M. 2009. The lemonade game competition. http://tech.groups.yahoo.com/group/lemonadegame/.