Enforcing Honesty in Assured Information Sharing Within a Distributed System

Ryan Layfield, Murat Kantarcioglu, and Bhavani Thuraisingham

The University of Texas at Dallas,
PO Box 830688, Richardson, TX 75083-0688
{layfield,muratk,bhavani.thuraisingham}@utdallas.edu
http://www.utdallas.edu/

Abstract. The growing number of distributed information systems such as the internet has created a need for security in data sharing. When several autonomous parties attempt to share data, there is not necessarily any guarantee that the participants will share data truthfully. In fact, there is often a large incentive to engage in behavior that can sabotage the effectiveness of such a system. We analyze these situations in light of game theory, a mathematical model which permits us to consider behavior and choices for any autonomous party. This paper uses this theory to create a behavior enforcement method that does not need a central management system. We use a simple punishment method that is inherently available in most scenarios. Our approach is applicable to a variety of assured information sharing applications where the members of a coalition have to work together to solve a problem.

1 Introduction

We live in an information-driven world where we use data from multiple information sources to solve problems. The local traffic report, for example, influences choices we make to get to work on time. However, there are several situations where the accuracy of information is crucial to success such as fighting a war or providing medical treatment. We need innovative ways to ensure that the accurate and secure data are shared between the parties. For example, the radio station we get our traffic data from is regulated by the FCC. Creating an honest environment when no central authority is available to enforce behavior presents a new set of challenges. Peers in distributed environments must therefore be sophisticated enough to evaluate the actions of each other and have a consensus of what they should all be doing. The results of these evaluations must be compared with some agreed upon common goal.

In peer to peer systems, a collection of peers is a number of parties that have their own data and want to share data amongst themselves. The goal of this system is to find a way to guarantee everyone ends up with an accurate corpus that reflects all of the data available. Traditionally, this has been done in a hybridized environment of peers and centralized management entities.

S. Barker, G.-J. Ahn (Eds.): Data and Applications Security 2007, LNCS 4602, pp. 113–128, 2007.

Despite the simplicity of their implementation, centralized systems have a number of drawbacks. First, this system must be trusted by all peers to be an impartial judge of activity. Second, such a system by design represents a single point of failure. They can be broken, attacked, and even compromised, effectively eliminating both the usefulness of the system and the services it offers. Third, they must be scaled when more peers want to join the system. In general, centralized management introduces several weaknesses into the data sharing environment. If we want to eliminate the need for such entities, we must find a way for the parties to trust each other.

Parties, however, may have only limited influence over each other. They have entered into this data sharing environment to gather data only because they can offer data. When parties have sufficient motivation to get more out of the system than they give up, the effectiveness of such a system must be questioned. Otherwise, if all parties attempted to cheat the system by sharing bad data in exchange for good data, none of the parties would gain anything.

1.1 Our Approach

Game theory, the study of competition and cooperation through the use of mathematics, offers a solution to this problem. These autonomous parties have no need to share good data since there is no incentive to do so. However, we can assume that they are rational and logical entities that are at least partially interested in collecting good data from the rest of the sources available. We can then assume proper coercion can illicit desirable behavior.

There are several options available in game theory which are designed to elicit behavior. One approach is to create a virtual economy in which data can be bought and sold for a price based on trust. While this is initially a promising solution, it turns out that balancing such an economy without a centralized regulatory entity proves unnecessarily complex [9]. Another option is to fight undesired behavior with undesired behavior: mimic the last data action performed, basically punishing a partner by mimicking their own actions. This works primarily when all data has the same value to all participants, a poor assumption for real-life scenarios. This method is known as Tit-for-Tat, and we discuss it in section 4.2.

We instead explore another alternative: non-participation. Consider a distributed sharing environment in which all of these participants are interested in acquiring each other's data. To keep information private, everyone establishes a secure individual connection with everyone else, forming a fully connected network. Information is traded over these connections simultaneously, where any two parties can exchange pieces of data. In the event that a party believes their trade partner has cheated during an exchange, the link is severed indefinitely. Thus, anyone who chooses to deviate from the desired actions will lose any potential gain from trading with that party in the future.

One of the biggest challenges most applications of game theory face in the real world is the assumption that we know exactly what the other players are doing. Perfect knowledge allows for robust decision making and more efficient choices.

For the sake of realism, we assume that knowledge of party actions comes at a price. Verification is the process of determining what a party has chosen as their actions, and we assign a cost to this process.

Another type of challenge is Trust Management. In a perfect world, everyone is trustworthy and never considers cheating the system or breaching security. The truth is, we deal with distrust due to malicious behavior on a regular basis. The goal of Trust Management is to decide whom to trust and how far we trust them. When we can determine this, dealing with sensitive information becomes much easier. However, as with the issue of perfect knowledge, we rarely know this beforehand. Therefore, a party that wants to properly maintain security must constantly evaluate their peers. In turn, when those evaluations indicate a party is untrustworthy, they must be punished. Punishment allows us to actually force a rational party into becoming more trustworthy by eliminating the benefit of being untrustworthy.

We have two objectives in our work. First, we want to determine the conditions in which non-participation punishment is effective. We use game theory to estimate the existence of these conditions and how they can be made. Secondly, we want to verify the existence of our results by running simulations using a model that simulates changes in behavior.

1.2 Motivating Scenario

Consider the international political environment. We have a number of countries, each with sovereign authority over the affairs of the state and their own set of interests. To protect those interests, we assume that each one has an intelligence agency designed to gather information in the interest of national security. Assume there is some event that, if it is allowed to occur, could threaten the security of any nation to which it happens. The problem these agencies face is that their data are limited. They may have field operatives working in other countries, but in most situations, they deal primarily with the affairs of their own country. If a threat emerged that spanned multiple international borders, it would be difficult for any one agency to track.

Given the severity of the threat, these agencies have decided to establish a mutual agreement: they will share the information they have gathered in exchange for information other agencies hold. Reality dictates that even if they have equal information resources, there is little incentive for agencies to share their real data or to keep the policies associated with it (i.e. secret classification). These organizations tend to reflect the relations that their respective countries have with each other. The agreement makes no provision for requiring any given policy to be enforced due to the lack of a common governing entity. Thus, there is no provision to prevent sabotage within the loose alliance; each agency must invest resources to know what the rest are doing.

Even with a fixed cost of discovering such information, there are a number of factors to consider. If an agency chooses to verify all of the actions taken by others, they will waste resources when their fellow agencies are behaving appropriately. However, if they become too trusting, other agencies can take

advantage of this situation without the fear of being caught. We assume that every agency at least has a basic security policy of not sharing bad or corrupted data. Now, consider the use of some punishment and assume that one of these agencies has decided to lie to everyone about recent reports on militia activity. Regardless of the motivation, when that agency is caught, our behavior dictates that they will be isolated from the rest of the data sharing network due to negligent and deviant behavior. Since the information of all agencies are roughly equivalent in value, the loss of this one data source will not affect the ability of the rest of the network to prevent such a militia from causing further trouble. However, when a sufficient number of participants ostracize the offending party, the agency that has been cut off is now left out of what is a valuable group effort. The isolation also serves as a warning to other agencies that may choose to deviate from providing quality information.

1.3 Related Work

Much of our work builds on the foundations of Agrawal et. al. [2]. That paper analyzed the issue of trust management among parties and proposed a solution that uses a management entity that 'taxes parties which use undesirable strategies. This 'tax' comes in the form of a discount on the gain in utility within a game matrix. Our work uses a similar model with two fundamental differences. First, we use a simple withdrawal strategy that terminates the game if bad strategies are used. Second, the responsibility of punishment is completely distributed to all of the parties, eliminating the need for a centralized manager.

In the realm of distributed systems, an area that has garnered considerable attention is that of peer-to-peer file distribution networks. The work here is aimed at enforcing trustworthy behavior in protocols such as BitTorrent [3] and distributed computing. Most of the work we found in this area considers only a third-party, but the works of [10], [5], and [2] deal explicitly with peer-based recourse for deviant behavior.

There is no shortage of game theory driven analysis on behavior enforcement. The work of [1] has inspired our approach to repeated games, but the general works of [5] and [11] have been notable in our efforts as well. None of this research to our knowledge, however, deals with the possibility of refusing to participate within a game. Instead, they suggest choosing a damaging strategy as a form of punishment for a specific amount of time.

Our current research is actually a refinement on our existing work in this field, published as a technical report [8]. We originally attempted to construct a behavior that could govern interaction in a hostile, purely peer-based game that provided the option of either lying or telling the truth. Trade in this case happened between two parties by their own choice, instead of all parties simultaneously. Punishment occurred when certain trade partners were favored more than others, leaving parties that chose to lie with less of a chance of being selected for trade.

1.4 Organization of This paper

Section 2 presents our approach to the game theory, payoff matrix, role of verification, and how malicious behavior is punished. Section 3 is a proof of the sub-optimality of our theory. We discuss how we tested our equilibrium in section 4 along with a detailed listing of significant competing behaviors. The results of our experiments are outlined in 5. Our conclusions, observations, and future directions are left to section 6.

2 Putting a Price on Consequence

We consider our scenario as an application of 2-person evolutionary game theory. The intelligence agencies are represented by game theory agents, which have behaviors and choose a strategy when deciding what to do each round. The measurement of an agency's success is determined by the amount of 'good' data that has been collected. To simplify the array of policies we can choose to enforce, we focus on a simple security policy of telling the truth. Thus, the strategies explicitly available are to Lie, tell the $Truth$, or $Withdraw$, but the option of verification also factors into how an agency can behave.

The value of information varies depending on who receives it and what context they plan to use it in. Data rarely has a uniform benefit to intelligence agencies in the real world. The perception some party i holds about the value of data in a particular round of trade t is Δ_t^i . This value is assumed to be bounded within some range, and represents the raw gain to the party receiving it.

Next, we must address the issue of verification. Using the data acquired during an exchange immediately will obviously cost less than spending resources to verify as long as the information is valid. Therefore, we associate a fixed cost with verification that is uniform among all parties for the sake of fairness. The cost of verification is represented by the constant C_V.

Always verifying results would ensure that the other party never succeeded in deviations such as lying, but it is wasteful with trustworthy parties. Therefore, the probability p_t^i that a single party i will verify the results in a round of transactions t should be inversely proportional to the probability that any given party will tell the truth. Note that no verification allows an attacker to build trust then betray it without consequence.

We assume that agencies have the ability to change their behavior at will. In most real-life situations, parties will periodically change their behavior if they believe it will help them. To accomplish this effect, we adopted the use of a genetic algorithm to allow behavior to 'evolve' among agencies. We save the explanation of this for discussion later.

Based on these observations, we have constructed a payoff matrix that reflects what every rational party should consider. The complete set of actions Γ available to each agency is $[Truth, Lie, Withdraw]$. We assume these actions are only considered on a per-interaction basis; that is, we only consider party strategy choice in pairs during trade.

		Player 1		
		Truth	*Lie*	*Withdraw*
Player 2	*Truth*	$\Delta_t^1 - p_V^2 C_V$ $\Delta_t^2 - p_V^1 C_V$	$\Delta_t^1 - p_V^2 C_V$ $-p_V^1 C_V$	0 0
	Lie	$-p_V^2 C_V$ $\Delta_t^2 - p_V^1 C_V$	$-p_V^2 C_V$ $-p_V^1 C_V$	0 0
	Withdraw	0 0	0 0	0 0

Fig. 1. Payoffs for each pair of strategies during trade

Variable	*Meaning*
Δ_t^i	The value of information offered by agent i in round t of the simulation
p_V^i	The probability that agent i will perform verification
C_V	The cost of performing verification

Fig. 2. The lookup table for variables used in figure 2

The $\{Truth, Truth\}$ strategy is trivial. Both parties expect to receive the utility value of the data from each other, minus the estimated cost of verification should they choose to do so. This is calculated by evaluating how often verification takes place.

Selecting $\{Truth, Lie\}$ or $\{Lie, Truth\}$ is where deviant behavior is introduced. Although we believe equilibrium is virtually impossible to achieve at these points, they must be evaluated: selection of these actions means an equilibrium does not exist yet. Consider two parties i and j that, up to round $k-1$, have been telling each other the truth. Both parties do not expect the other to tell a lie, and as such the probability of verification p_{k-1}^i is at the minimum threshold. It becomes possible therefore in round k that i could lie to j with little chance of being caught. By doing so, i gains the value of j's shared data without ever having to give up significant data of their own, giving i an immediate advantage. If this action is performable without being caught over long periods of time, i can guarantee that they will gain more data and ultimately decrease the effective amounts of information j can acquire.

However, if the choice is $\{Lie, Lie\}$, neither party gains anything. Both parties waste resources for taking the time to interact. Any verification within an equilibrium of this behavior would only add to the loss. In the real-world, this would likely mean that the organization would only benefit if they withdraw from the alliance entirely.

This is the point at which we consider *Withdraw* as an option. When played, the party severs their link with another party, eliminating any further trade. Such an action should be considered a last resort. For example, i has chosen to withdraw from it's connection with j, it will from that point on no longer gain anything from j in future rounds of the game. This would be a tremendous loss, negating any future gains for either party. Therefore, since any strategy choice

with $Withdraw$ in it has the same results, $\{Withdraw, Withdraw\}$ becomes an automatic (and undesirable) Nash equilibrium.

Verification has become an interesting factor in the success of behavior. Always choosing to verify would decrease the overall benefit of trade when dealing with a highly reliable source. Reflecting periodic verification checks in the matrix would unnecessarily complicate the game theory and would require a much more complex model.

One of the most interesting characteristics of our application of game theory is the uncertainty of the other party's actions. The nature of the information we consider is not easily verifiable. Most of the research in the area of data sharing does not address games with imperfect information. We believe that reflecting such a property in our work makes our research much more practical, especially when discovering such perfect information comes at a measurable utility cost in the real world.

3 Equilibrium Emergence

Before analyze the above game, we briefly introduce some of the related game theoretic notions.

For any vector $v = (v_1, \ldots, v_n)$, we use v_{-i} to represent $(v_1, \ldots, v_{i-1}, v_{i+1}, \ldots v_n)$, and (v_i, v_{-i}) to denote the reconstruction of the v.

Definition 3.1 ***Nash Equilibrium****[1] A strategy profile $\sigma^* = (\sigma_1^*, \sigma_2^*)$ is a Nash equilibrium in a two person game with utility functions u_i if the following inequality hold for each agent i,*

$$u_i(\sigma_i{}^*, \sigma_{-i}{}^*) \geq u_i(a_i, \sigma_{-i}{}^*)$$

where a_i belongs to set of possible actions A_i that could be taken by agent i

Intuitively, the above definition states that if all agents predict that a particular equilibrium will occur then no player has an incentive to deviate from equilibrium strategy.

Consider a traditional one-shot game. We must pick a strategy in which we can guarantee our success. Consider $Withdraw, Withdraw$ as a natural Nash equilibrium. At first glance, this would appear to be a poor choice. Clearly, better payoffs are found in $Truth, Truth$. However, if we choose $Truth$ as our strategy of choice in this setup, the other player can choose Lie as it increases their utility. If we choose Lie instead, we can take advantage of another player's trust. Should they choose $Truth$ and deviate from the equilibrium, their payoff will dramatically decrease while ours increases; at Lie, our payoff is as we expected. $Withdraw$ of course neutralized both results. Thus, a Nash equilibrium exists at Withdraw,Withdraw.

In practice, not all games are classified as one-shot. Some involve players that play the same game multiple times. Such games enable players to use past data to both predict their opponent's behavior and even affect a particular outcome.

In our model, the "data sharing" game will be played many many times by the participating agents. This scenario can be easily modeled by the "repeated game" ideas from game theory literature [6]. The main observation in repeated games is that the honest behavior in games like the "data sharing" game can be enforced if the game continues to be played with probability $\delta > 0$. In other words, if there are possible future gains, (i.e. if game continues with some probability) each agent can be motivated to be truthful.

We can define the expected payoff for a player i participating in the repeated "data sharing" game as the

$$u_i = (1-\delta)\sum_{t=0}^{\infty} \delta^t \cdot g_i({\sigma_i}^t, {\sigma_{-i}}^t)$$

where $\sigma^t = (\sigma_i^t, \sigma_{-i}^t)$ is the strategy employed at time t, δ is the halting probability of the game, and g_i is the gain achieved at each play of the "data sharing" game. Let $u = (v_1, v_2)$ be the payoff vector of the repeated game. Note that if every period $g_i({\sigma_i}^t, {\sigma_{-i}}^t)$ is equal to some u then u_i will be equal to u.

To illustrate, consider an instance of the game between two intelligence agencies a_1 and a_2 at some point in time on round t. From the perspective of a_1, ${\sigma_{-i}}^t$ is expected to be $Truth$ for a_2 since ${\sigma_{-i}}^{t-1}, {\sigma_{-i}}^{t-2}, \ldots, {\sigma_{-i}}^1$ have all been $Truth$ as well. According to this equation, we should expect the maximum utility of u for $Truth, Truth$. However, a_1 could have a behavior that tries to deviate at round t if a_2 has proven trustworthy. In this instance, σ_i will be Lie, and v will be greater than $Truth, Truth$.

Below we prove that our repeated "data sharing" game can be used to enforce truthful behavior by refusing the deal with dishonest agents that caught cheating. Our proof technique is very similar to the one used for proving "Nash Folk" theorem from the repeated game theory literature [6]. Our main difference as compared to the generic Nash Folk theorem is that in our case opponents actions could not be observed unless a party to choose to verify the correctness of the data. Given the above "data sharing" game, we can prove that truth telling emerges as a Nash equilibrium as follows:

Theorem 3.1 *If telling the truth each round has a gain $g_i > 0$ for both parties then there exits $0 < \delta < 1$ such that telling the truth for both parties is a Nash Equilibrium for "data sharing" game.*

Proof. **Sketch**
We will prove that utility of telling the truth given that the other party tells the truth is bigger then any other strategy that lies with some probability p. To see that let us calculate the expected gain of a given party who chooses to lie with probability $p > 0$ at each round. Note that in a given round with probability $(1-p)$ he will gain $g_{T,T}$ (i.e. the gain achieved when both party tells the truth) and with probability p he will gain $g_{L,T}$ (i.e.the gain achieved when he lies while the other party is telling the truth). If he cheats and is caught, he will earn zero for the rest of the game; otherwise, a new round starts. Under these observations,

we can write the total expected utility of lying with probability p given that the other party verifies the correctness of the received data with probability q as

$$u_i = (1-p) \cdot g_{T,T} + p \cdot g_{L,T} + (1 - p \cdot q) \cdot \delta \cdot u_i \tag{1}$$

$$= \frac{(1-p) \cdot g_{T,T} + p \cdot g_{L,T}}{1 - (1 - p \cdot q) \cdot \delta} \tag{2}$$

Similarly we can write the utility of always telling the truth (denoted as u_i^T below) if the other party tells the truth as

$$u_i^T = g_{T,T} + \delta \cdot u_i^T \tag{3}$$

$$= \frac{g_{T,T}}{1-\delta} \tag{4}$$

Note that $u_i^T > u_i$ if we set the δ such that it satisfies the following inequality

$$\delta > \frac{\frac{g_{L,T}}{g_{T,T}}}{\frac{g_{L,T}}{g_{T,T}} - q - 1}$$

Therefore, for the above given δ, telling the truth will be a Nash equilibrium because each party has no incentive to lie given that the other party is telling the truth.

4 Simulation Construction

Obviously, if every party used the game theory we proposed as their primary logic, we would have no issue with reaching an equilibrium immediately. However, we would instead like to see how our design interacts in a variety of game environments. A diverse environment will enhance the robustness of our theory.

The gaming environment of our design is straightforward. We have a collection of N game theory agents representing parties that interact via secure bidirectional communication pipes. Each party a_i is initially linked to every other party in the system, forming a fully connected graph. This pipe can be broken voluntarily by the party at either end. We assume this pipe is completely secure from tampering or eavesdropping for the sake of simplicity. All players act simultaneously in each round.

We wanted to analyze the results of our theoretical conclusions in a diverse environment of party behaviors. In order to do this, we use three existing possible approaches to this scenario ($Random, Tit\text{-}For\text{-}Tat, Dishonest$), our own behavior $Truthful\text{-}Punisher$ along with two variations on our own work ($Liar, SubtleLiar$).

4.1 Random Behavior

The *Random* behaviors simply randomly selects Lie or $Truth$. This strategy represents a party which has no desire to spend time on the details of the alliance

while simultaneously lacking a consistent motivation to adopt proper behavior. In theory, this randomized behavior can succeed when other parties do not consider the past and there is little effect due to punishment.

4.2 Tit-for-Tat Behavior

Next, we have the famous *Tit-For-Tat* strategy. A party using this strategy starts by telling the truth. After that, this party mimics whatever action was taken by their trade partner. Research has proven that, unless other parties conspire against it in some fashion, this is the most effective behavior possible for games resembling the Prisoner's Dilemma, as discovered by Anatol Rapoport [4].

Within our scenario, this behavior operates at a disadvantage. Since perfect information is not free, the party must verify the results of each and every trade they make. This could lead to a situation in which it actually gains less utility against behaviors that are relatively trustworthy but have little regard for verification.

However, it also has a potentially larger advantage: it does not use the grim trigger punishment system. The idea behind punishment is that the party takes a calculated "hit" to the immediate trade benefits by refusing to deal with parties that do not tell the truth, in the hopes that they will become more honest. While this has obvious ramifications for dishonest behavior, unless interacting with that party has a net loss (i.e. tells a lie more often than it tells the truth), it is still beneficial to maintain an open communications channel and choose the less harsh strategy of mimicking their choice. In essence, this behavior should provide the best competition to our own construction.

4.3 Dishonest

In order to add the appropriate amount of realism to our scenario, we must also consider parties that have no desire to contribute meaningfully to the group. Such behaviors are simply classified as *Dishonest*, and as such they choose to always lie. They still may reap the benefits of those that choose to tell them the truth, but they will never bother to verify what they receive nor punish those that lie as well. Thus, this agent exists in our simulation solely to insure the rest of the parties cannot make the assumption that all behaviors will ultimately yield any sort of positive or 'break-even' net gain. This is in contrast with the *Random* behavior, which will arguably still yield a net gain of zero through prolonged participation.

4.4 Truthful-Punisher Behavior

Before we describe the variations on our ideal behavior, we must first describe what our game theory analysis has suggested to us. Since there is a clear Nash Equilibrium at $\{Truth, Truth\}$ with our punishment modifications, our behavior always chooses $Truth$. The probability of verification is done as a percentage that is handed off as part of the behavior characteristics. When the simulation is first

created, each time a party using this behavior type (or a variant) is instantiated, a random percentage is chosen for its' verification probability. Essentially, this party either tells the truth or cuts the other party off.

4.5 Periodic Liar Behavior

The first variant on our behavior is to try and get away with lying a fraction of the time. This is designed to represent an party that believes they can deviate from time to time when they have a desire to sabotage their competition. More importantly, it simply represents a mindset in which the party does not believe that the original conclusions of always telling the truth is a true equilibrium within the 'real-world' environment, making it a close relative of the *Random* behavior.

4.6 Subtle Liar Behavior

In theory, any party could choose to deviate only when they know that their trade partner is going to give them valuable data. They believe they can lie without worry of significant punishment. We assume that party i will choose $\{Lie\}$ during communication with j whenever $\Delta_t^i > \Delta_T$ during round t, where Δ_T is simply a threshold above a significant majority of all possible piece values. This is especially handy when dealing with $Tit\text{-}For\text{-}Tat$, as retaliatory behavior assumes that by lying to them on the next round will neutralize gains from deviation. Since piece values vary over a set range, this works to the behavior's advantage as long as the data it will not receive due to punishment during the next round is of lesser value.

5 Experiments

Information is exchanged between our virtual parties every round. During each round, a party trades with the rest of the parties they are connected to. No one party has an advantage over the other through knowledge of the move their partner has made due to the synchronous nature of our setup.

All experiments are run to no more than 20,000 rounds. The game will terminate early if an equilibrium is achieved (i.e. all agencies go to a particular behavior, leaving no other possible behaviors to choose from). Note that we are not explicitly using game theory approaches for infinitely repeated games; we assume that at some point there will no longer be a need for the alliance.

We judge a party's fitness by the value of accumulated data. Each time a party is told the truth, the data value is added to the total value of the party. Whenever a party chooses to verify, the cost of verification is subtracted. Obviously, being told a lie and choosing to verify the data will result in a net loss. In the spirit of our scenario, a positive gain in data is much more desirable.

Every L rounds, we pause the game to perform an evaluation of the performance of these parties. This reflects when agencies choose to evaluate their

performance to maximize efficiency. First, we need to see how each of them has performed since the last check. Every party's gain q_i is calculated from the increase in their net value. This value is added up to yield a total utility value over the whole system, q_T.

Next, we calculate $p^i_{behavior} = q_i/g_T$ for each party's behavior, where $p^i_{behavior}$ denotes the normalized probability that the behavior held by party i should be used by parties in the next generation. A behavior embodies both the core logic mentioned in section 4 along with any attributes. We use this normalized percentage as a way of measuring how well a particular approach has performed in contrast with the rest of the system. The higher a party's relative gain, the higher their percentage 'score'.

We want our population to reflect the fitness of each behavior proportionally, according to the basics of genetic algorithms[12]. In our scenario, we assume that agencies will want to maximize their data trading success by adopting the behaviors of those that are most successful. Since $\sum_{i=0}^{n} p^i_{behavior} = 1$, we can use $p^i_{behavior}$ to ensure that the next 'generation' of agencies adhere to this principle. We thus reassign the strategies for every agency based on this probability. We do not consider the very real notion that a successful agency is unlikely to want to change it's strategy; we simply need the population to reflect the evolved characteristics of the system.

We expect a number of properties to emerge based on our analysis. First, we expect that our equilibrium-based behavior to outperform and ultimately dominate the overall population given enough generations. Next, we expect that this behavior will adjust it's verification rates based on how many parties use a deviant behavior. Finally, we expect our approach to dominate all possible variations available.

5.1 Results

The outcome of our experiments confirms our theory is correct. Verification and punishment appear to be highly effective even in a diverse population, as our simulation consistently converged to a homogeneous population of our particular behavior. There's a clear correlation between the use of our punishment method and the success of agents within the system. Since agencies that did not obey the truth policy were cut-off, agencies which told the truth remained within a somewhat exclusive clique. As long as this clique's benefits exceeded those that are offered by those outside of it, the system eventually began to encompass only agencies that used the adhered to an honest strategy remained after just a few rounds.

As we suspected, $Tit\text{-}For\text{-}Tat$ did not perform well enough to beat our strategy. Despite the fact that the strategy kept links open to several parties which still offered a net gain, the population eventually became devoid of any dishonest participants. Once this happened, $Truthful\text{-}Punisher$'s ability to settle for less than perfect information (via infrequent verifications) gave it a clear advantage, as fewer resources were wasted verifying information that would never be a lie. It at times took several generations, but eventually, $Tit\text{-}For\text{-}Tat$ would disappear from our population, leaving only $Truthful\text{-}Punisher$ with complete dominance.

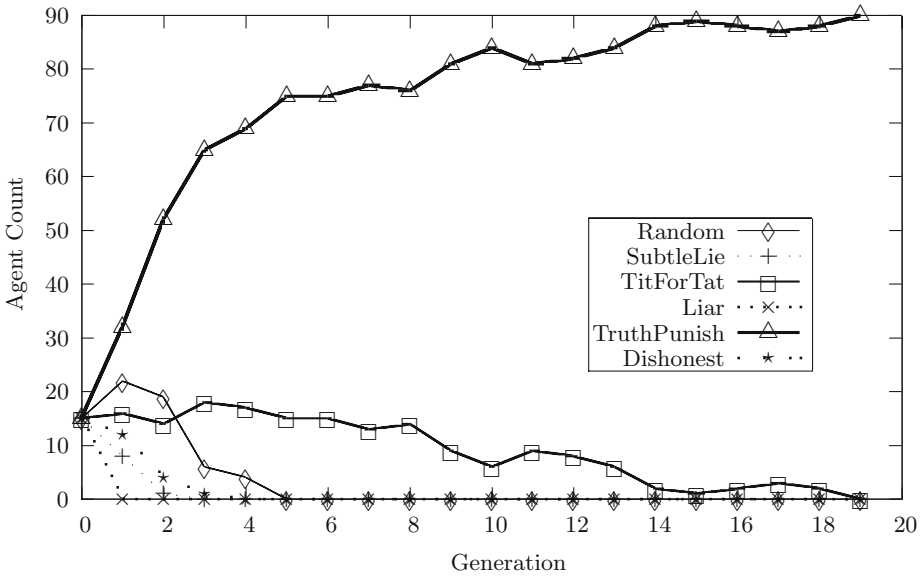

Fig. 3. Behaviors by population per generation

The rest of the behaviors which choose not to tell the truth consistently did not perform well enough to pose a threat to our equilibrium. Despite the two variants which attempted to lie only periodically, those that choose such an approach were eventually caught and collectively punished. The only time this did not happen was when we set a very low threshold for the SubtleLiar behavior; when the probability of lying was less than 10%, the difference made to the agency's performance was so low that our experiments converged to SubtleLie with equal probability. However, since the net gain from such periodic lying was also very low, we suspect that this has more to do with a small fraction of difference lost in the varying cost of the pieces. Additionally, the probability that they would be caught was simply too low to be of significant value. At rates at or above this threshold, our original behavior prevailed consistently against it.

Convergence to our behavior happened in an average of 20 generations. The leading competitor often ended up clinging to a small portion of the population as a handful of agents before eventually succumbing to the agents bearing our behavior. Typically, this was a small sub-population of no more than 5 parties, which were usually *Tit-For-Tat*. We believe the reason for this is rooted in the fact that our own constructed behavior tends to seek a verification probability based on how honest the population is at a given time. As our behavior propagates, the need to verify decreases, leaving it more vulnerable to future attacks that never occur.

One of the ways in which we observed system performance is by way of four metrics: *Truces*, *Fools*, *Follies*, and *DeadLinks*. *Truces* represent both parties choosing to tell the truth. *Fools* are situations in which one party told the truth

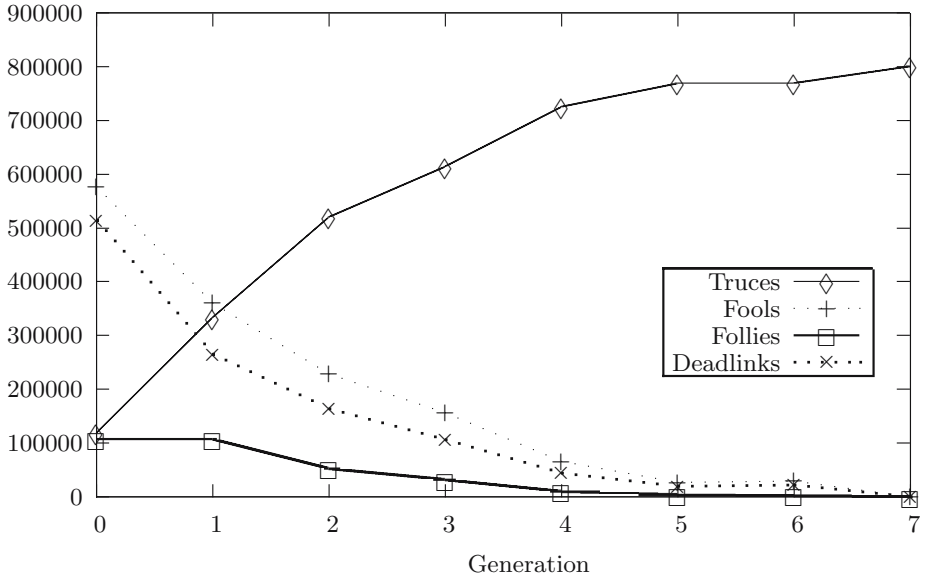

Fig. 4. Chosen strategy pairs per generation

while the other lied. *Follies* represent when both parties choose to lie to each other, resulting in either no gain or net loss. *DeadLinks* simply indicated when trade never happened between two parties. In Figure 4, we see that *Truces* exceed the other options in only four generations, indicating that the system is achieving optimality as early as possible.

The dynamic behavioral choices of any given iteration of our simulation involve the life and death of the various behavioral choices, as seen in Figure 3. Starting from an equal distribution of six different behaviors, we find that the first variant *Liar* 'dies' after only one generation. The next death is of the *SubtleLiar* variant in the generation following. Both of these variants are surprisingly beaten by the *Random* behavior; clearly, even small deviations from our proposed behavior can prove disastrous. Finally, and least surprisingly, is the delayed demise of *Tit-For-Tat*, which is the last behavior to go. This is most likely due to the fact that it will never lie to a trustworthy opponent which punishes; thus, it is never isolated from the productive members of the group and only loses due to dependency on perfect information.

6 Conclusions

Overall, we were pleased with the results of our simulations. The populations converged rather quickly to a $\{Truth, Truth\}$ equilibrium, and our behavior eventually overcame any competition provided there was enough time. Dishonest behaviors were eliminated rather quickly, even in variants of the experiments we performed with unfair advantages given to competing behaviors. Although their

were restrictions in involved on its' effectiveness, we were overall pleased to say that our work has fulfilled our objectives.

Despite this success, we were not entirely satisfied with the results. Clearly, there is no need to continue verifying results once the system converged to a Truth strategy. We originally asserted that the verification rate would thus go to our lower bound for verification (10%), as agents using a strategy reflecting little or no verification should pull ahead. Instead, our system simply approached a 30% rate with significant deviation. The problem we believe lies in the nondeterministic nature of our choice of genetic algorithms. Since verification costs are relatively small compared to the payoff from the information, there is always a net gain regardless of verifying the information when the truth is always told. However, even when we doubled the cost of verification and set payoff in $\{Truth, Truth\}$ to be a constant, there was simply never enough of a gain to converge.

The last question we want to answer is how effective our agent is as a group. Obviously, a single agency cutting off others is not going to be a significant deterrent on their own. Our results show that approximately 40% of the population must use punishment to significantly deter others from deviating from $Truth$. This reflects similar findings found in distributed computing, such as the Byzantine Generals Problem, in which a certain majority of the participants must be trustworthy in order to properly succeed against deviant strategies [7].

Another pressing issue is the vulnerability of the population to constructed behaviors which could wait for convergence to a mostly honest population and then switch to a dishonest policy of strategy choice. Since the characteristics of our design favors a more vulnerable state when it appears 'safe' to do so, we are concerned that future generations would have little defense against a growing dishonest population. The only way we could combat this is to introduce mutation rates among our behavioral characteristics.

The most endearing application of our work is how it can apply to the enforcement of any desired behavior. The nature of the Folk theorem is that, with sufficient patience and time, any desirable equilibrium can be achieved. Based on this work, we can enforce any security policy as long as the actions taken are verifiable in some capacity. Given the traditional approaches that require a management party, true distribution of responsibility makes it possible to have much more robust security that does not rely on any one entity to enforce behavior.

In order to bring more realism to our model, our future work will also address two major assumptions: imperfect verification and insecure lines of communication. The former deals with situations in which we cannot guarantee information will be properly classified as a truth or a lie. The latter raises the possibility that we have insecure communication channels; a would-be attacker could easily cause communication disruption through information tampering. Both can be addressed by assigning a confidence factor in the form of a probability reflecting the likelihood that data can be trusted while relaxing the grounds on which the $Withdraw$ option is selected.

References

1. Agarwal, N.: Equilibrium Game Theory Under the Conditions of Repeatability. SSRN eLibrary (2002)
2. Agrawal, R., Terzi, E.: On honesty in sovereign information sharing. In: Ioannidis, Y., Scholl, M.H., Schmidt, J.W., Matthes, F., Hatzopoulos, M., Boehm, K., Kemper, A., Grust, T., Boehm, C. (eds.) EDBT 2006. LNCS, vol. 3896, Springer, Heidelberg (2006)
3. Andrade, N., Mowbray, M., Lima, A., Wagner, G., Ripeanu, M.: Influences on cooperation in bittorrent communities. In: P2PECON '05. Proceeding of the 2005 ACM SIGCOMM workshop on Economics of peer-to-peer systems, New York, NY, USA, pp. 111–115. ACM Press (2005)
4. Axelrod, R.: The Evolution of Cooperation. Basic Books (1985)
5. Buragohain, C., Agrawal, D., Suri, S.: A game theoretic framework for incentives in p2p systems. In: P2P '03. Proceedings of the 3rd International Conference on Peer-to-Peer Computing, Washington, DC, USA, p. 48. IEEE Computer Society (2003)
6. Fudenberg, D., Tirole, J.: Game Theory. MIT Press, Cambridge, Mass (1991)
7. Lamport, L., Shostak, R., Pease, M.: The byzantine generals problem. ACM Trans. Program. Lang. Syst. 4(3), 382–401 (1982)
8. Layfield, R., Kantarcioglu, M., Thuraisingham, B.: Research and simulation of game theoretical techniques for data sharing among semi-trustworthy partners. Technical Report UTDCS-46-06, The University of Texas at Dallas (2006)
9. Mahajan, R., Rodrig, M., Wetherall, D., Zahorjan, J.: Experiences applying game theory to system design. In: PINS '04. Proceedings of the ACM SIGCOMM workshop on Practice and theory of incentives in networked systems, pp. 183–190. ACM Press, New York, NY, USA (2004)
10. Monderer, D., Tennenholtz, M.: Distributed games: from mechanisms to protocols. In: AAAI '99/IAAI '99. Proceedings of the sixteenth national conference on Artificial intelligence and the eleventh Innovative applications of artificial intelligence conference innovative applications of artificial intelligence, Menlo Park, CA, USA, pp. 32–37. American Association for Artificial Intelligence (1999)
11. Myerson, R.: Game Theory: Analysis of Conflict. Harvard University Press, Cambridge, Mass (1991)
12. Riolo, R., Worzel, B.: Genetic Prgoramming Theory and Practice. Kluwer Academic, Boston (2003)