Evolving Defensive Strategies Against Iterated Induction Attacks in Cognitive Radio Networks

Siming Liu, Shamik Sengupta, and Sushil J. Louis
Department of Computer Science and Engineering, University of Nevada, Reno
1664 N. Virginia Street, Reno Nevada, United States
simingl@cse.unr.edu, ssengupta@unr.edu, sushil@cse.unr.edu

Keywords: Genetic Algorithm, Induction Attack, Cognitive Radio Networks

Abstract: This paper investigates the use of Genetic Algorithms (GAs) to evolve defensive strategies against iterated and memory enabled induction attacks in cognitive radio networks. Security problems in cognitive radio networks have been heavily studied in recent years. However, few studies have considered the effect of memory size on attack and defense strategies. We model cognitive radio network attack and defense as a zero-sum stochastic game. Our research focuses on using GAs to recognize attack patterns from different attackers and evolving defensive strategies against the attack patterns so as to maximize network utility. We assume attackers are not only able to attack high utility channels, but are also capable of attacking based on the history of high utility channel usage by the secondary user. In our simulations, different memory lengths are used by the secondary user against memory enabled attackers. Results show that the best performance strategies evolved by GAs gain more payoff, on average, than the Nash equilibrium. Against our baseline memory enabled attackers, GAs quickly and reliably found the theoretically globally optimal defensive strategy. These results indicate that GAs is a viable approach for generating strong defenses against arbitrary memory based attackers.

1 INTRODUCTION

A cognitive radio is an intelligent radio that detects available channels in wireless spectrum and accordingly adapts its transmission and reception parameters to operate in these unused channels with other devices. Traditional methods of spectrum allocation and management are strictly followed by regulatory organizations like the Federal Communications Commission (FCC) in the United States. However, most radio frequency spectrum are not efficiently utilized. Cellular network bands are overloaded but other frequency bands such as television broadcasting bands and amateur radio bands are not fully utilized. Therefore, new technologies for improving spectrum utilization has become a popular focus of attention for next generation wireless networks. In cognitive radio networks, a licensed user is the primary user who could use the specified band anytime, while an unlicensed user is a secondary user who could use the specific band whenever the primary user is not using the band.

However, flexibility makes cognitive radio networks vulnerable and easy to attack. Primary User Emulation (PUE) is an effective way to attack cognitive radio, and is usually called “jamming” in traditional wireless networks. Machine learning algorithm attack is another type of attack which compromises the ability of secondary users to learn the environment. This type of attack leads the secondary user to learn misleading temporal and spatial characteristics of the radio frequency environment and adapt their functionalities and configurations to sub-optimal or even a malicious strategy. The characteristic of being intelligent enables cognitive radios to learn their environment and learn an optimal strategy to maximize the utility of the network, but this ability to learn may also expose the radio to spurious environment information and thus the learning of incorrect behavior. For example, some dynamic spectrum access algorithms gather channel access statistics for channel usage rate to predict the utility of a channel. If attackers deceive the channel usage information on a channel, it will deceive the secondary user about the utility of this channel and cause the secondary user to choose a low utility channel to transfer data. It is also called induction attack. Figure 1 shows how a cognitive radio change its beliefs and behavior based on manipulated sensor input.

Practically, attackers usually have their own char-
Figure 1: Manipulated environment information input can change the beliefs and behavior of a cognitive radio.

characteristics, personalities, purposes, and experience (memory). For example, some attackers want to maximize their own data throughput in a selfish manner. Others intend to minimize the spectrum utility of the network and maximize the damage to legitimate users. In iterated attack and defense games, memory enabled attackers attack the radio network with different attacking patterns. For example, one type of attacker may consider blocking the high utility channels causes bigger problem. Some attackers may think blocking the history channels that the secondary user used deals more damage to the radio network. Other attackers may combine the previous two attacking strategies. We are interested in applying techniques which are capable of recognizing attackers’ attacking patterns and then taking advantage of their weakness to maximize the spectrum usage of the secondary users. In this paper, we used GAs to evolve counter strategies for different types of attackers. We focus on the memory enabled induction attack in a cognitive radio network and propose a stochastic game framework for defense design, which can accommodate channel quality and both the secondary users and attackers’ history of moves. We modeled the induction attack between secondary users and attackers in cognitive radio networks as an iterated zero-sum stochastic game. Instead of considering the real utility of each channel, we use the rank of all the available channels at the end of each time slot. Then we sort our channels based on the channel utility rank and consider the top 3 highest utility channels. Given the memory length $N$, we encoded each strategy as a $9^N \times 2$ bit-string, and repeatedly run this strategy against an attacker 3000 times to calculate the average payoff of this strategy. We applied GAs to search for effective strategies against different types of attackers. The results show that our GA could recognize the characteristics of attackers and evolve optimal counter strategies to maximize the average payoff.

The remainder of this paper is organized as follows. Section 2 describes related work in security research and common techniques used in cognitive radio networks. In section 3, we introduce the system model for the secondary users, the attackers, and the available channels. Section 4 describes our simulation environment and parameter encodings for the GAs. Section 5 presents preliminary results and compares the quality, reliability, and the time taken to find solutions. Finally, section 6 draws conclusions and discusses future work.

2 RELATED WORK

A lot of work has been done in applying variant artificial intelligence techniques to cognitive radio networks. Artificial Neural Networks (ANN) have been applied to spectrum sensing, radio parameter adaptation, and other aspects of cognitive radio (Haykin, 1994). Fehske et al. developed an ANN based signal classifier utilizing distinct features of each signal type which were extracted using cyclic spectral analysis (Fehske et al., 2005). Cattoni et al. used an ANN to classify different IEEE 802.11 signals based on frequency features (Cattoni et al., 2007). Zhu et al. proposed a channel sensing algorithm based on adaptive resonance theory neural networks for wireless mesh networks (Zhu et al., 2008). GAs, which were inspired from natural selection and genetic evolution of species in nature, have been widely used to solve multi-objective optimization problem and dynamically configure cognitive radio networks (Goldberg, 1989). Rondeau et al. introduced an adaptive cognitive radio component, which uses GAs to evolve a radio with a certain set of parameters and maximize the spectrum usage (Rondeau et al., 2004). Newman et al. presented a GA driven, cognitive radio decision engine that determines the optimal radio transmission parameters for single and multi-carrier systems (Newman et al., 2007). Park et al. proposed a novel optimization algorithm named goal-pareto based non-dominated sorting GA and validated its applicability on a CDMA2000 forward link in a realistic scenario (Park et al., 2007). We are interested in applying GAs to finding effective strategies for induction attack and defense scenarios within cognitive radio networks.

Considering research into vulnerabilities in cognitive radio networks, Bhattacharjee et al. classified various types of vulnerabilities and provided an overview of the research challenges (Bhattacharjee et al., 2013). They categorized five different types of attackers and studied the effects of different genetic
operators. Then they described the current research advances in countering these categorized attacks in cognitive radio networks. Brown et al. examined the denial of service vulnerabilities in cognitive radio networks and explored potential protection remedies that can be applied (Brown and Sethi, 2008). Their results showed that the advantage attackers gained from jamming can be significantly reduced with modest effort on the part of the cognitive radio design. They presented a multi-dimensional analysis to highlight relative risks for each combination of CR network architecture, spectrum awareness method, and spectrum access method.

Many researchers worked on applying game theory to analyze the security of cognitive radio networks. Manshaei et al. provide a structured and comprehensive overview on the application of game theory in addressing the security and privacy problems in computer networks (Manshaei et al., 2011). They reviewed various game theoretical formulations of network security issues and outlined the security problems with corresponding game theoretical approaches. Zhu et al. introduced a stochastic zero-sum (Markovian) game model for a secondary user and an attacker in a jamming and anti-jamming scenario (Zhu et al., 2010). Their results showed that the payoffs of the secondary users increase with the number of available jamming-free channels and are eventually limited by the behavior of primary users. Li et al. modeled primary user emulation attacks as a dogfight game in spectrum between defending secondary users and an attacker (Li and Han, 2010; Li and Han, 2011). They modeled the dogfight in spectrum as a zero-sum game and calculated the Nash equilibrium. They also applied their framework to the case of multi-stage dogfight by fixing the defense strategy of secondary user.

Axelrod used a GA to evolve new strategies in the iterated prisoner’s dilemma (Axelrod, 1987). He encoded three previous moves as short-term memories of players in iterated prisoner’s dilemma game. He successfully evolved five behavioral patterns that mirrored what TIT FOR TAT would do in similar circumstances. Phelps et al. combined evolutionary optimization together with a principled game-theoretic analysis to automatically acquire strategies for the double-auction market problem (Phelps et al., 2006). Tahk et al. used the augmented Lagrangian approach to transform a constrained optimization problem to a zero-sum game with the saddle-point solution, and then applied a co-evolutionary algorithm to solve the problem. (Tahk and Sun, 2000). James worked on developing a biologically inspired model of cognition in a radio architecture and introduced a Wireless Channel Genetic Algorithm, Wireless System Genetic Algorithm, and Cognitive System Monitor architectures and algorithmic framework (Rieser, 2004).

Wang et al. proposed a stochastic game framework for anti-jamming defense in cognitive radio networks (Wang et al., 2011). They used minimax-Q learning for secondary users to learn the optimal policy which maximizes the expected sum of spectrum-efficient throughput. Wang’s research is closest to our work in this paper. We are interested in applying evolutionary computing algorithms to solve a memory enabled attack and defense game between the secondary user and attackers in cognitive radio networks modeled as an iterated zero-sum game. Specifically, we are interested in applying GAs to finding effective defensive strategies for a zero-sum stochastic game modeled on the security aspect of cognitive radio networks. To the best of our knowledge, this is the first work using GAs to evolve optimal defensive strategies in a memory enabled iterated stochastic game modeled from a security scenario in cognitive radio networks.

3 SYSTEM MODEL

In this section, we present the assumptions for modeling induction attack and defense between secondary users and malicious attackers in cognitive radio networks.

We consider a cognitive radio system with $N$ secondary users and $L$ licensed channels. Each secondary user can sense only one channel at the beginning of each time slot. Spectrum sensing errors are not considered in our work. In this preliminary research, we consider only one secondary user and one attacker at a time in our induction attack and defense scenario. Channels can be used by the secondary user for data transmission when they are not occupied by primary users. We assume that both the secondary user and attacker have perfect information, that is, they can perfectly sense all channel information at the end of each time slot. Each licensed channel is modeled as a random process and we model the induction attack as an iterated stochastic game. A stochastic game is an extension of a Markov Decision Process (MDP). In a stochastic game $G$, there is a set of states $S$, a set of actions of the secondary user $A_s$, a set of actions of the attacker $A_a$. The game is played in a sequence of stages. Each stage will transit to a new stage with transition probability of $T : S \times A_s \times A_a \mapsto P(S)$. Each player receives a payoff $R_i : S \times A_s \times A_a \mapsto \mathbb{R}$. The expected payoff of each player is: $E \{ \sum_{j=0}^{\infty} r_{i,f+j} \}$, where $j$ is the number
utility channels by certain patterns. These patterns will be considered by the secondary user and the attacker. Table 1 shows the payoff matrix of the iterated stochastic game between the secondary user and the attacker.

We assume the lower utility channel as much as possible, but also trying to block one previous step of the secondary user. In this case, we assume this attacker attacks the highest utility channel \( C_1 \) 70% of the time, attacks the previous step of the secondary user with 30% chance.

- **Defensive Attacker**: This attacker attacks only the history steps of the secondary user. Their goal is to block the secondary user as much as possible based on their history moves. In our experiments, we assume the defensive attacker attacks the previous step of the secondary user 44.44% of the time, attacks the second previous step 33.33% of the time, and attacks the third previous step 22.22% of the time.

- **Moderate Attacker**: This attacker combines both attack patterns from the defensive attacker and the aggressive attacker. The attacker want to block high utility channels as well as blocking the secondary user. We assume the attacker attacks \( C_1 \) 50% of the time, attacks the previous step of the secondary user 25% of the time, and attacks the second previous step 25% of the time as well.

According to the listed assumptions above, we know that the secondary user aims to maximize the network utilization with the evolved channel switching strategies against different types of attackers.

### 4 METHODOLOGY

A GA uses mechanisms that mimic the process of natural biological evolution, such as reproduction, mutation, and selection. In this paper, we apply our GAs in a context of induction attack and defense between the secondary user and the attacker in cognitive radio networks.

There are various methods of representing each allowable defense strategy as a bit-string chromosome which can be used by GAs. In our study, we consider the set of strategies as deterministic and use the outcomes of the previous \( N \) moves to make a choice in the next move. According to Table 1, giving 9 possible outcomes for each move, there are \( 9^N \) different histories of the \( N \) previous moves. In the case of 3 steps of memory, there are \( 9^3 = 729 \) different histories. Figure 2 shows how a strategy can be specified by an ordered list of 729 \( C_1, C_2 \), and \( C_3 \)'s. Therefore, our secondary user in the simulation can calculate the list index from the previous 3 moves and chooses the next move to be the corresponding channel at the calculated list index. The chromosome thus
Table 2: Chromosome length and search space.

<table>
<thead>
<tr>
<th>History</th>
<th>Chromosome</th>
<th>Search Space</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>9 × 2 = 18</td>
<td>3^9 = 16983</td>
</tr>
<tr>
<td>2</td>
<td>81 × 2 = 162</td>
<td>3^81 × 2 = 4.43 × 10^{38}</td>
</tr>
<tr>
<td>3</td>
<td>729 × 2 = 1458</td>
<td>3^729 × 2 = 6.63 × 10^{347}</td>
</tr>
<tr>
<td>4</td>
<td>6561 × 2 = 13122</td>
<td>3^6561 × 2 = 2.47 × 10^{1130}</td>
</tr>
</tbody>
</table>

represents a given strategy. Three channels need 2 bits to represent. Then the chromosome length becomes 729 × 2 = 1458. Figure 2 shows list index, represented histories and a sample strategy.

Figure 2: Strategy representation.

To store the history moves during the simulation, it is necessary to specify an initial premise about the hypothetical moves in the chromosome. Given the memory length of 3, we need 6 more genes for the secondary user and the attacker, making a total of 1458 + 6 × 2 = 1470 bits. The 6 premise genes will be initialized by 6 hypothetical moves at the beginning of the game, and will be updated at the end of each move in the simulation. Therefore, the string of 1470 bits would serve as the individual’s chromosome for use in genetic operations.

Detailed chromosome lengths and search spaces of secondary users with different memory length are shown in Table 2. Considering 3 steps of memory, we can see that the search space is 3^9 = 6.63 × 10^{347}. The exhaustive search algorithm for effective strategies in this huge dataset is clearly not efficient enough. To find optimal or near-optimal strategies in such a huge collection we need a different technique. We used a GA in our simulation to search for optimal defensive strategies.

We evaluated a chromosome by simulating a game consisting of 3000 moves and calculating the accumulated payoff based on the payoff matrix. Our GA used CHC elitist selection instead of the canonical roulette wheel selection (Holland, 1975; Eshelman, 1991). This type of selection allows offspring and parents to compete for population slots in the next generation. More specifically, our GAs select the P best individuals out of both parents and offspring to create the next generation, where P is the population size. Early experiments indicated that our elitist selection GA worked significantly better than the canonical GA on our problem. For the genetic operators, we set the population size to 300 and ran the GA for 450 generations. The probability of crossover is 70% and we used CHC selection. We also used bit-mutation with 0.1 chance of each individual bit flipping in values. CHC being strongly elitist selection that keeps valuable information from being lost if our GA produces low fitness children. Algorithm 1 shows the GA used in our simulations.

Algorithm 1 Genetic Algorithm

1: Initialize population
2: for each of 450 generations do
3:   for each of 300 individuals do
4:     for each of 3000 moves do
5:       SecondaryUserMove();
6:     AttackerMove();
7: UpdateScore();
8: end for
9: end for
10: Reproduce the next generation
11: Selection
12: Crossover
13: Mutation
14: end for

5 RESULTS AND DISCUSSION

In this section, we conduct simulations to evolve an optimal defensive strategy for the secondary user with different lengths of memory against different attackers. Our simulation environment is an Intel Xeon X5660 dual core CPU with 6 GB memory. We run our GAs with ten different random seeds, which enables us to obtain and report on statistically significant results. According to the payoff matrix, we can calculate that the payoff of the Nash equilibrium strategy is 0.516 (Osborne, 1994). With a mixed strategy of 32.26% C1, 22.58% C2, 45.16% C3. Therefore, this is a game which is slightly biased towards the secondary user, and the secondary user will gain 0.516 against reasonable attackers.

We first consider evolving strategies for the secondary user’s defense against the Random Attacker, who randomly attacks the three highest utility channels. The highest performance strategy found by our
GA is using $C_1$ 100% of the time, and the payoff of the secondary user is 0.847. This is theoretically the optimal strategy for the secondary user’s defense against the Random Attacker. Since the secondary user cannot predict the next move of the Random Attacker, choosing the highest utility channel to transfer data will be the optimal strategy. The average payoff through 3000 consecutive moves of the secondary user with 3 steps of memory is 0.914, which is higher than the payoff of the Nash equilibrium point 0.516.

Against the Aggressive Attacker, the GA found that the global optimal strategy for the secondary user is switching between $C_2$ and $C_3$. The aggressive attacker will attack the highest utility channel 70% of the time, and attack the previous step of the secondary user the remaining 30% of the time. The 70% chance of attack on $C_1$, lowered the utility of this channel. With the 30% probability of attack on the secondary user’s previous channel, the secondary user learned to hop between channels $C_2$ and $C_3$, which effectively dodged the attacker. The results also indicates that the secondary user only needs the same length of memory as the attacker and successfully learned the optimal strategy, which got the high payoff of 2.5. Longer memory doesn’t help the secondary user to improve the performance against the attacker with only one step of memory. Therefore, the secondary user with two or three steps of memory found the same optimal strategy, but took longer than the attacker with only one step of memory because the search space for the longer memory representation is much larger than the shorter one.

The Defensive Attacker attacks all the previous moves of the secondary user. With only one step of memory, the attacker deterministically attacks the previous step of the secondary user. The results show that the optimal strategy found by our GA is hopping between $C_1$ and $C_2$. This strategy not only takes the two highest utility channels, but also dodges the attacker from blocking the previous used channel. The payoff of this strategy is as high as 3.5. For the Defensive Attacker with two steps of memory, the optimal strategy found by our GA is hopping from $C_1$ to $C_2$, $C_2$ to $C_3$, and then back to $C_1$. Since the attacker attacks the last two steps of the secondary user, the third previous channel is safe to use. The Defensive Attacker with 3 steps of memory, attacks all top 3 utility channels used by the secondary user, there is no pure strategy to dodge the attacker. Therefore, the GA found a mixed strategy with 38.23% to choose $C_1$, 32.9% to choose $C_2$, and 28.86% to choose $C_3$. The payoff of this strategy, 1.167, is doubled than the Nash equilibrium.

The Moderate Attacker combines the attack patterns from the Aggressive Attacker and the Defensive Attacker. This attacker with one step of memory behaves the same as the Aggressive Attacker. Table 3 shows that the optimal strategy against this attacker is the same as the optimal strategy used against the Aggressive Attacker, and the payoff is again 2.5. The Moderate Attacker with two steps of memory may attack $C_1$, $C_2$, and $C_3$ channels. No pure strategy dominates other strategies. Our results show that the GA found strategy produces a payoff of 1.4987 for the secondary user with two steps of memory, and 0.825 with three steps of memory.

Table 3 shows the payoffs and the best performance strategies found by the secondary user with the same length of memory as different attackers. The same strategies may get slightly different payoffs be-

---

Table 3: Best performances and strategies between secondary users and attackers.

<table>
<thead>
<tr>
<th>Attacker</th>
<th>AT Mem</th>
<th>SU Mem</th>
<th>Payoff</th>
<th>Best Strategy</th>
</tr>
</thead>
<tbody>
<tr>
<td>Random</td>
<td>0</td>
<td>1/2/3</td>
<td>1.019</td>
<td>All $C_1$</td>
</tr>
<tr>
<td>Aggressive</td>
<td>1</td>
<td>1/2/3</td>
<td>2.501</td>
<td>50%$C_2$, 50%$C_3$, Switch between $C_2$, $C_3$.</td>
</tr>
<tr>
<td></td>
<td>1</td>
<td>2</td>
<td>3.5</td>
<td>50%$C_1$, 50%$C_2$, Switch between $C_1$, $C_2$.</td>
</tr>
<tr>
<td></td>
<td>2</td>
<td>2</td>
<td>3.001</td>
<td>33.3%$C_1$, 33.3%$C_2$, 33.3% $C_3$, Switch among $C_1$&amp;$C_2$, $C_3$.</td>
</tr>
<tr>
<td></td>
<td>2</td>
<td>3</td>
<td>1.167</td>
<td>38.23%$C_1$, 32.9%$C_2$, 28.86%$C_3$.</td>
</tr>
<tr>
<td>Defensive</td>
<td>1</td>
<td>2</td>
<td>2.501</td>
<td>50%$C_2$, 50%$C_3$, Switch between $C_2$, $C_3$.</td>
</tr>
<tr>
<td></td>
<td>2</td>
<td>2</td>
<td>1.499</td>
<td>32.43%$C_1$, 34.73%$C_2$, 32.83%$C_3$.</td>
</tr>
<tr>
<td></td>
<td>2</td>
<td>3</td>
<td>0.825</td>
<td>36.57%$C_1$, 32.93%$C_2$, 30.5%$C_3$.</td>
</tr>
</tbody>
</table>

Figure 3: The optimal strategy payoffs of the secondary user’s defense against different attackers with the same length of memory.
cause the random numbers generated by our simulator are not evenly distributed. For example, the Random Attacker generated 3000 moves should equally distribute to all three channel, but the result may contain 977 $C_1$, 992 $C_2$, 1031 $C_3$.

We modeled the induction attack as an iterated stochastic game. Instead of considering each channel’s utility, we use the rank of all available channels at the end of each time slot. Then we sort our channels based on the rank and consider the 3 highest utility channels. We encoded each strategy as a $9^N \times 2$ bit-string, and repeatedly run the simulation between this strategy and the fixed attacker 3000 times, then calculate the average payoff of this strategy. The results show that our GA can recognize the different attacking patterns of attackers and search for optimal strategies to adapt to the specific attacker. For example, against the Defensive Attacker who always attack the previous step of the secondary user, the GA found the strategy of switching from $C_1$ to $C_2$ to gain the maximum payoff from the network. Against the Aggressive Attacker, who attacks $C_1$ 70% of the time and the previous step of the secondary user 30% of the time, the GA found the optimal strategy of switching from $C_2$ to $C_3$.

We also evaluated how the different lengths of memory affect GA performance in finding counter strategies. The results show that longer memory provides more complicated strategies both from the secondary user and the attacker, but is not guaranteed to produce higher performance. For example, the Defensive Attacker with two or three steps of memory is able to evolve high payoff strategies against attackers with five or ten steps of memory.

However, the GA used in this study requires heavy computational power. One run of our GA needs around 10 minutes on average on our simulation platform. We can use smaller population size and number of generations, which could significantly decrease the calculation time down to one or two minutes. However, although these less computationally expensive parameters lead to GA performance that comparable to our prior results for “simple” attacking patterns like the Aggressive Attacker, their performance is not

6 CONCLUSION AND FUTURE WORK

Security concerns and the vulnerabilities of cognitive radio networks are still an open research area. In our study, we assume that not all attackers attack with the same optimal strategy. They usually have their own characteristics and personalities. In this paper, we used GAs to evolve counter strategies against different attacking patterns from the secondary user’s perspective.

We updated our simulations so that the secondary user has two or three steps of memory, but the attacker possesses longer memory. We added five and ten steps of memory to the Defensive Attacker. The attacker is able to attack the previous five or ten channels used by the secondary user with equal probability. For example, the attacker could attack any of the previous five channels with 20% chance, or any of the previous ten channels with 10% chance. Figure 4 shows the optimal strategy payoffs of the secondary user with 5 or 10 steps of memory.

We are also interested in scenarios where the attacker has longer memory than the secondary user. We used GAs to evolve counter strategies against longer memory attackers. We updated our simulations so that the secondary user with only two or three steps of memory defense against the attackers with 2 or 3 steps of memory. The results show that the secondary user with only two or three steps of memory can also evolve high payoff strategies against longer memory attackers.

<table>
<thead>
<tr>
<th>Attacker</th>
<th>SU Mem</th>
<th>AT Mem</th>
<th>Payoff</th>
<th>Best Strategy</th>
</tr>
</thead>
<tbody>
<tr>
<td>Defensive</td>
<td>2</td>
<td>5</td>
<td>1.242</td>
<td>34.87%$C_1$, 30.77%$C_2$, 33.37%$C_3$</td>
</tr>
<tr>
<td></td>
<td>2</td>
<td>10</td>
<td>1.223</td>
<td>33.47%$C_1$, 28.2%$C_2$, 38.33%$C_3$</td>
</tr>
<tr>
<td>Defensive</td>
<td>3</td>
<td>5</td>
<td>0.844</td>
<td>39.93%$C_1$, 28.13%$C_2$, 31.93%$C_3$</td>
</tr>
<tr>
<td></td>
<td>3</td>
<td>10</td>
<td>0.721</td>
<td>41.23%$C_1$, 30.07%$C_2$, 28.7%$C_3$</td>
</tr>
</tbody>
</table>

Figure 4: The optimal strategy payoffs of the secondary user with 2 or 3 steps of memory defense against the attackers with 5 or 10 steps of memory.

Table 4: Best performances and strategies between the secondary user and attackers with longer memory.
as good as with the larger population sizes on the complicated attack patterns. The second limitation is that our representation of strategies is unsuitable for long-term memory. Even with four steps memory for the secondary user, the search space is $2.47 \times 10^{130}$, which may take too long to search effectively.

We are also interested in evolving strategies from the attacker’s point of view that could maximize the damage to the network. In addition, we next plan to co-evolve the attack and defense strategies from both the secondary user’s and the attacker’s perspectives.

ACKNOWLEDGEMENTS

This research was supported by NSF grant IIA-1301726 and NSF CAREER grant CNS-1346600.

REFERENCES


