Numerical Computation of Multi-goal Security Strategies

Stefan Rass[1] and Benjamin Rainer[2]

[1] Universität Klagenfurt, Institute of Applied Informatics, System Security Group, Klagenfurt, Austria
stefan.rass@aau.at
[2] Universität Klagenfurt, Institute of Information Technology, Klagenfurt, Austria
benjamin.rainer@aau.at

Abstract. Security is often investigated in terms of a single goal (e.g., confidentiality), but in practical settings mostly a compound property comprising multiple and often interdependent aspects. *Security strategies* are behavior profiles that guarantee some performance *regardless* of how the adversary really behaves (provided that it stays within its action set). While security strategies towards a single goal are easy to compute via Nash-equilibria (or refinements thereof), playing safe towards multiple security goals induces the notion of Pareto-optimal security strategies. These were recently characterized via Nash-equilibria of multi-player games, for which solution algorithms are intricate and may fail for small instances already. Iterative techniques, however, exhibited good stability even for large games. In this work, we thus report on theoretical and practical results how security strategies for multiple (interdependent) goals can be computed via a set of simple transformations and a final application of humble fictitious play.

Keywords: Pareto-optimality, security strategies, game theory, equilibrium, fictitious play, security.

1 Introduction

Security strategies have been introduced in [18], as a mean of optimizing behavior under uncertainty of the opponent. That is, a security strategy gives the best payoff for player 1 under arbitrary, especially not equilibrium, behavior of player 2 in a two-person game. This models situations in which only the opponent's action space is known, but the player remains uncertain about the other's payoff structure(s). Information security is a natural incarnation of this, as we seek the optimal defense against arbitrary actions of an adversary, whose possible actions are known, but nothing about its particular behavior can be assumed reasonably. Treating a single security goal in that sense yields scalar two-person games in the style "honest-vs-adversary". However, most practical settings require simultaneous defense strategies against various different threats, such as violations

R. Poovendran and W. Saad (Eds.): GameSec 2014, LNCS 8840, pp. 118–133, 2014.

of confidentiality, integrity, availability and authenticity (CIA+ security). Security strategies accounting for simultaneously optimal payoffs in various perhaps strongly interdependent goals have not been studied very extensively so far, and are subject of this work.

Security strategies in the scalar case, i.e., when only a single security goal is of interest, are easily identified as Nash-equilibria of zero-sum games. In a multi-dimensional case, i.e., for security in multiple possibly interdependent aspects, Pareto-optimal security strategies are sought. Applications of these are manifold, e.g., creating high-security communication lines that are confidential, robust and authentic, can be achieved by multipath-transmission and multipath-authentication, which in turn leads to straightforward game-models (an example is given in section 6.2).

Searching for security strategies is interesting from a theoretical and practical point of view, as it can provide quantitative risk estimates. For example, setting up a transmission channel between to peers by virtue of multipath transmission, the game can be defined with the sender acting as player 1, who chooses the transmission configuration (in particular the paths over which information is conveyed). Player two is the adversary, who chooses nodes to attack. The game's payoff function is the fraction of correctly delivered messages, where "correctly" here covers confidentiality and integrity (at least). Given a particular network infrastructure (topology), what is the likelihood of achieving the two security goals upon a single transmission? The answer lets the sender utilize the network in a proper way so as to minimize the risk of security breaches, and can be used to enhance the network infrastructure (by additional protections at the most likely targets for the opponent (adversary) in the network infrastructure). So, the practical aspect of game-theory in network security is related to *topological vulnerability analysis*, where the competition between the (honest) network users and the adversary points out best practices to use the network, as well as neuralgic spots being indicated as the most likely attack strategies for the adversary (opponent player 2). We revisit this use case later.

Our focus here is, however, not on game-theoretic models of applied cryptography, but rather on covering a numerical problem in the computation of Pareto-optimal security strategies. These can be computed to support or enhance processes of topological vulnerability analysis and quantitative risk management. Especially the latter may call for efficient updates following changes (enhancements) to the system. Therefore, the efficiency of computing security strategies may be of interest besides its theoretical value.

In fact, relying on the characterization as obtained in prior literature (and cited below), "standard" algorithms to compute Nash-equilibria may be applied. Unfortunately, however, the whole armory of algorithms that ships with the Gambit software [10], rapidly failed to compute the sought results even for small examples (numerical instabilities occurred already in example instances with, e.g., three goals and eight strategies per player). On the bright side, fictitious play exhibited good numerical stability (though slow convergence) and has been proven capable of computing the sought security strategies even for large

games like those arising in our example application of security risk assessment of multipath communication channels.

2 Preliminaries

Throughout this work, normal font denotes scalars and bold face font denotes vectors. Sets are written in upper-case latin letters like N. The cardinality of a set N is denoted by $|N|$.

A *game* is a triple $\Gamma = (N, S, H)$, where N is the – in our case always finite – set of $n = |N|$ players, $S = \{PS_1, \ldots, PS_n\}$ contains the individual action sets for each player, and $H = \{u_1, \ldots, u_n\}$ is the family of payoff functions $u_i : \prod_{i=1}^{n} PS_i \to \mathbb{R}$ for each $i \in N$.

As a standard shorthand notation, we write PS_{-i} for the cartesian product of all $PS_j \in S$, excluding PS_i. The vector $(s_1, \ldots, s_{i-1}, s_{i+1}, \ldots, s_n) \in PS_{-i}$ is abbreviated as $\boldsymbol{s}_{-i}$.

Hereafter, we write s for pure strategies, but mostly consider mixed strategies, i.e., probability distributions over the action sets. For simplicity, we thus denote S_i as the set of all probability distributions supported on a set PS_i of actions, also called *pure strategies*. This is the set of *mixed* strategies. Such mixed strategies and general probability distributions are denoted by lowercase Greek letters, e.g., $\theta, \phi \in S_i$. We will hereafter drop the attribute "mixed", as we will not explicitly talk about pure strategies any more (and because pure strategies arise via degenerate mixed strategies anyway). Random variables are denoted by uppercase letters like X; their distribution θ is told by the symbol $X \sim \theta$.

A *Nash-equilibrium* in an n-person game is a set of strategies $(\theta_1^*, \ldots, \theta_n^*)$ so that all players $i \in N$ receive for all $\theta_i \in S_i$ an expected payoff $\mathsf{E}_{(\theta_i^*, \theta_{-i}^*)} u_i(X_i, X_{-i}) \geq \mathsf{E}_{(\theta_i, \theta_{-i}^*)} u_i(X_i, X_{-i})$, where the expectation is taken over the probability distributions noted in the subscripts of the expectation operator. By a slight abuse of notation for the sake of simplicity, we let $u_i(\theta_i, \phi_i)$ also denote the long-run average payoff (over an infinite number of repetitions of the game[1]), as we will exclusively speak about expected payoffs in in the context of mixed strategies. In that notation, the Nash-equilibrium condition in a two-person zero-sum game (expected payoff functions being u_1 and $-u_1$) can compactly be written as

$$u_1(\theta, \phi^*) \leq u_1(\theta^*, \phi^*) \leq u_1(\theta^*, \phi) \quad \forall \theta \in S_1, \phi \in S_2, \tag{1}$$

where the pair (θ^*, ϕ^*) denotes the equilibrium, and we call $v = u_1(\theta^*, \phi^*)$ its *(saddle-point) value*.

[1] Even if the game cannot be repeated, then using indicator variables for the payoffs turns the expected payoffs into probabilities. In this setting, the Nash-equilibrium is the likelihood to win (or loose) in a single round of the game, thus making the concept applicable even if the game is not repeatable.

3 Security Strategies

Towards an axiomatic characterization of security strategies in general games (finite or infinite), captured as definition 1, we take known results in the scalar case as the template for upcoming definitions.

3.1 The Scalar Case

The following is a well-known fact (cf. [2] among others).

Lemma 1. *Let $\Gamma = (N, S, H = \{u_1, u_2\})$ be a two-person game with continuous payoff functions. Define the zero-sum game $\Gamma_0 = (N, S, H_0 = \{u_1, -u_1\})$, with Nash-equilibrium $v = u_1(\theta^*, \phi^*)$. Then, player 1 always receives $u_1(\theta^*, \phi) \geq v$ in Γ, no matter how player 2 actually behaves. Moreover, there is a strategy $\phi' \in S_2$ so that $u_1(\theta^*, \phi') = v$ in Γ.*

The lower bound provided by the zero-sum equilibrium value is easily obtained by observing that player 2 due to a perhaps different payoff structure in Γ most likely deviates from the optimal zero-sum strategy ϕ^* in Γ, thus leaving player 1 with more than the zero-sum equilibrium payoff v. The existence of a strategy ϕ' achieving equality directly follows from the continuity of the payoff functions.

The ordering of $\mathbb{R}$ that lets us define the equilibrium condition is lost upon the transition to $\mathbb{R}^k$ for $k > 1$. This unfortunate fact renders the proof of lemma 1 non-transferable to $\mathbb{R}^k$, and calls for more sophisticated concepts.

3.2 The Multi-criteria Case

A *multi-objective game* (MOG) has vector-valued payoffs. That is, the i-th player receives r_i different payoffs, denoted by the function $\boldsymbol{u}_i : \prod_{i=1}^{n} PS_i \to \mathbb{R}^{r_i}$, $(s_i, \boldsymbol{s}_{-i}) \mapsto (u_i^{(1)}(s_i, \boldsymbol{s}_{-i}), \ldots, u_i^{(r_i)}(s_i, \boldsymbol{s}_{-i}))$. For two vectors $\boldsymbol{a} = (a_1, \ldots, a_k)$, $\boldsymbol{b} = (b_1, \ldots, b_k) \in \mathbb{R}^k$, we write $\boldsymbol{a} \leq \boldsymbol{b}$, if $a_i \leq b_i$ for all $i = 1, 2, \ldots, k$. The complement relation is $\boldsymbol{a} >_1 \boldsymbol{b}$ and holds iff an index $1 \leq j \leq k$ exists such that $a_j > b_j$, no matter what the other components do. The vector-relations $\geq, <_1, \leq_1$ and $\geq_1$ are defined accordingly.

The sibling of Nash-equilibrium in the scalar case is the Pareto-Nash equilibrium in the multivariate case: here, we require the inequalities in (1) to fail in at least one component upon a deviation from the optimum. That is, an n-player MOG $\Gamma = (N, S, H)$ admits a *Pareto-Nash equilibrium* $(\theta_1^*, \ldots, \theta_n^*)$ if for every player $i \in N$, we have $\boldsymbol{u}_i(\theta_i, \theta_{-i}^*) \leq_1 \boldsymbol{u}_i(\theta_i^*, \theta_{-i}^*)$ for every θ_i. For two players, the resulting pair of inequalities resembles the equilibrium condition (1) by requiring that optimality fails in at least one goal by any deviation from the Pareto-Nash strategy profile $(\theta_i^*, \theta_{-i}^*)$.

In [13], a precursor definition towards an axiomatic characterization of *network provisioning* security strategies is given. We adapt this construction into our definition 1 here that is not confined to problems of secure data delivery.

Definition 1. *A strategy $\theta^* \in S_1$ in a two-person multi-criteria game Γ with continuous payoff $\boldsymbol{u}_1 : S_1 \times S_2 \to \mathbb{R}^k$ for player 1, is called a* multi-criteria security strategy *(MCSS)* with assurance vector $\boldsymbol{v} = (v_1, \ldots, v_k)$*, if the following two conditions hold:*

1. *The assurances are the component-wise guaranteed payoff for player 1, i.e. for all components i, we have*

$$v_i \leq u_1^{(i)}(\theta^*, \phi) \qquad \forall \phi \in S_2, \tag{2}$$

   with equality being achieved by at least one choice $\phi_i \in S_2$.
2. *At least one assurance becomes void if player 1 deviates from $\boldsymbol{x}^*$ by playing $\theta \neq \theta^*$. In that case, some $\phi \in S_2$ exists such that*

$$\boldsymbol{u}_1(\theta, \phi) \leq_1 \boldsymbol{v}. \tag{3}$$

Observe that the above definition transforms the assertions of lemma 1 in the scalar case into axioms in the multi-dimensional case. The existence of multi-dimensional security strategies has been studied in the literature, where the following characterization was established:

Theorem 1 ([13]). *Let Γ be a two-player MOG. The distribution θ^* constitutes a multi-criteria security strategy (MCSS) $\boldsymbol{v}$ for player 1 and k goals in the game Γ, if and only if it is a Pareto-Nash equilibrium strategy for player 0 in the following $(k+1)$-player multi-objective* auxiliary game $\overline{\Gamma} = (N, S, H)$*, where: $N = \{0, 1, \ldots, k\}, S = \{PS_1, PS_2, \ldots, PS_2\}$ (i.e. a multiset with $|S| = k+1$) and the payoffs are $\overline{\boldsymbol{u}}_0(s_0, \ldots, s_k) := (u_1^{(1)}(s_0, s_1), \ldots, u_1^{(k)}(s_0, s_k))$ for player 0 (vector-valued), and $\overline{u}_i(s_0, \ldots, s_k) := -u_1^{(i)}(s_0, s_i)$ (scalar-valued) for the opponents $i = 1, 2, \ldots, k$.*

From theorem 1, the existence of security strategies is not immediately evident, but can be concluded from results of [9] concerning the existence of Pareto-Nash equilibria in multiobjective games (MOG).

Theorem 2 ([9]). *Let $\Gamma = (N, S, H)$ be a MOG, where each $PS_i \in S$ is convex and compact, and each $\boldsymbol{u}_i \in H$ is continuous. Moreover, assume that for each player $i \in N$, every individual payoff $u_i^{(j)}(s_i, \boldsymbol{s}_{-i})$ for $1 \leq j \leq r_i$ is a concave function of s_i on PS_i, whenever the remaining values $\boldsymbol{s}_{-i}$ are fixed. Then, Γ has a Pareto-Nash equilibrium.*

From this we easily obtain the existence of MCSS under various conditions. For example, every finite game admits multi-criteria security strategies, which re-proves a known result of [1] by a humble application of theorems 1 and 2:

Corollary 1 (Existence of MCSS in matrix games). *Every finite MOG admits a multi-criteria security strategy.*

We will not go into further details about existence of MCSS, beyond stressing the fact that definition 1 is not limited to finite games or games with a finite number

of players. In that sense, the characterization theorem 1 can be obtained with alternative results to theorem 2 to establish the existence of MCSS for various other classes of games.

For simplicity, e.g. security risk management in multipath communication networks, we can work with corollary 1 to handle the arising matrix-games.

The proof of theorem 2 is "constructive" in the sense of equating the set of Pareto-Nash equilibria to the set of Nash-equilibria in a scalarized version of the MOG. Specifically, [9] prescribe the following steps to find a Pareto-Nash equilibrium in the n-player MOG Γ:

1. Fix an arbitrary set of real numbers $\alpha_{11}, \alpha_{12}, \ldots, \alpha_{1r_1}, \alpha_{21}, \ldots, \alpha_{2r_2}, \ldots, \alpha_{n1}, \ldots, \alpha_{nr_n}$ that satisfy condition (4):

$$\left.\begin{array}{ll} \sum_{\kappa=1}^{r_i} \alpha_{i\kappa} = 1 & \text{for } i = 1, 2, \ldots, n, \text{ and} \\ \alpha_{i\kappa} > 0 & \text{for } \kappa = 1, 2, \ldots, r_i \text{ and } i = 1, 2, \ldots, n. \end{array}\right\} \tag{4}$$

2. Form a (scalar) game $\Gamma_s = (N, S, H')$ with $H' = \{f_1, \ldots, f_n\}$ and

$$f_i = \sum_{\kappa=1}^{r_i} \alpha_{i\kappa} u_i^{(\kappa)}. \tag{5}$$

3. Find a Nash-equilibrium $\theta^* = (x_1^*, \ldots, x_n^*)$ in Γ_s, which is then a Pareto-Nash equilibrium in Γ.

Notice that the Nash-equilibria found by the above algorithm depend on the particular choice of weights. Indeed, the full set of equilibria is given as the union of all equilibria over all admissible choices of α's in (4) [9].

4 Numerical Computation of MCSS

Although there exist sophisticated algorithms and implementations to compute Nash-equilibria in multi-person games, an experimental implementation of our transformation using the GAMBIT software [10] showed that these algorithms fail on games with many players and strategies. It therefore appears advisable to prefer iterative numeric techniques over analytic ones for practical settings, in which we can expect a large number of strategies and security goals, the latter of which correspond to players. Our method of choice is fictitious play.

4.1 Fictitious Play in Multi-criteria Compound Games

Briefly speaking, fictitious play is the process of repeatedly playing the game while every player notes and learns the other player's moves, while at the same time optimizing his/her own behavior based on the so-far recorded behavior profiles. More concretely, let $t \in \mathbb{N}$ be the sequence of discrete time steps. Player i moves along a sequence of actions $(s_i(t))_{t\in\mathbb{N}} \in PS_i$ and maintains *beliefs* for each opponent $j \neq i$ that are discrete probability distributions for each $t \in \mathbb{N}$ of

the form $\left(\beta_i(t) = \frac{1}{t}\sum_{\tau=1}^{t}\delta_i(\tau)\right)_{i=1}^{k}$. Here, δ_i is the Dirac probability distribution that assigns unit mass to action s_i (by this convention PS_i is included in S_i as extremal points). Player i's next move at time $t+1$ is then the optimal response to its recorded opponent behavior profile $(\beta_i^1(t),\ldots,\beta_i^{i-1},\beta_i^{i+1},\ldots,\beta_i^n(t))$ at time t. We say that a game has the *fictitious play property*, if this process approaches an equilibrium θ^* in the sense that for every $\varepsilon > 0$ there is some t_0 such that for every $t \geq t_0$, we have $\left\|(\beta_i^1(t),\ldots,\beta_1^n(t)) - \theta^*\right\| < \varepsilon$ in some norm. See [17] for a more comprehensive account.

4.2 Computing MCSS by Fictitious Play

In the terminology of [17], the auxiliary game $\overline{\Gamma}$ is a "one-against-all" multi-player game or *compound game*, which can be solved iteratively by fictitious play if it were zero-sum. Although theorem 1 specifies $\overline{\Gamma}$ not as zero-sum, this can be fixed easily without changing the set of equilibria. Indeed, it is the scalarization (5) that will become helpful in a twofold manner, as it lets us apply standard fictitious play and it lets us prioritize our security goals.

Given a two-player MOG Γ and its auxiliary game $\overline{\Gamma}$, we prepare the latter for fictitious play by making it zero-sum before the necessary scalarization. To this end, recall that player 1 in Γ, who is player 0 in $\overline{\Gamma}$, has k goals to optimize, each of which is represented as another opponent in the auxiliary game $\overline{\Gamma}$. We define the payoffs in a *compound game* ("one-against-all") from the payoffs in $\overline{\Gamma}$, while making the scalar payoffs vector-valued to achieve the zero-sum property:

– player 0:

$$\overline{\boldsymbol{u}}_0 : PS_1 \times \prod_{i=1}^{k} PS_2 \to \mathbb{R}^k,$$
$$\boldsymbol{u}_0(s_0,\ldots,s_k) = (u_1^{(1)}(s_0,s_1), u_1^{(2)}(s_0,s_2),\ldots,u_1^{(k)}(s_0,s_k))$$

– i-th opponent for $i = 1,2,\ldots,k$:

$$\overline{\boldsymbol{u}}_i = (0,0,\ldots,0,-u_1^{(i)},0,\ldots,0). \tag{6}$$

Obviously, the "vectorization" of the opponents payoffs does not affect any equilibrium conditions, so the so-modified game comes with the same set of equilibria as $\overline{\Gamma}$. To numerically compute (one of) them, we scalarize as follows: to each of player 0's k goals, we assign a weight $\alpha_{01},\ldots,\alpha_{0k}$. The scalarization in (5) is via

$$\alpha_{ji} := \alpha_{0i} \text{ for } i = 1,2,\ldots,k \text{ and } j = 1,2,\ldots,k.$$

With these weights, the payoffs in the scalarized compound game are:

– for player 0: $f_0 = \alpha_{01}\overline{\boldsymbol{u}}_1 + \alpha_{02}\overline{\boldsymbol{u}}_2 + \cdots + \alpha_{0k}\overline{\boldsymbol{u}}_k$,
– for the i-th opponent, where $i = 1,2,\ldots,k$

$$\begin{aligned} f_i &= \alpha_{01}\cdot 0 + \alpha_{02}\cdot 0 + \cdots + \alpha_{0,i-1}\cdot 0 + \alpha_{0i}\cdot(-u_1^{(i)}) + \alpha_{0,i+1}\cdot 0 + \alpha_{0k}\cdot 0 \\ &= -\alpha_{0i}\cdot u_1^{(i)} \end{aligned} \tag{7}$$

Concluding the transformation, we obtain a scalar compound game

$$\overline{\Gamma}_{sc} = (\{0, 1, \ldots, k\}, \{PS_1, \underbrace{PS_2, \ldots, PS_2}_{k \text{ times}}\}, \{f_0, \ldots, f_k\}) \tag{8}$$

from the original two-person MOG Γ with payoffs $u_1^{(1)}, \ldots, u_1^{(k)}$ that can directly be be plugged into expressions (6) and (7).

Towards a numerical computation of equilibria in $\overline{\Gamma}_{sc}$, we need yet another transformation due to [17]: for the moment, let us consider a general compound game Γ_c as a collection of k two-person games $\Gamma_1, \ldots, \Gamma_k$, each of which is played independently between player 0 and one of its k opponents. With Γ_c, we associate a two-person game Γ_{cr} that we call the *reduced game*. The strategy sets and payoffs of player 0 in Γ_{cr} are the same as in Γ_c. Player 2's payoff in the reduced game is given as the *sum* of payoffs of all opponents of player 0 in the compound game.

Lemma 2 ([17]). *A fictitious play process approaches equilibrium in a compound game Γ_c, if and only if it approaches equilibrium in its reduced game Γ_{cr}.*

So, it suffices to consider the reduced game $\overline{\Gamma}_{scr}$ belonging to $\overline{\Gamma}_{sc}$. It is a trivial matter to verify the following fact (by substitution).

Lemma 3. *The reduced game $\overline{\Gamma}_{scr}$ of the scalarized compound game $\overline{\Gamma}_{sc}$ defined by* (8) *is zero-sum.*

So by the famous result of [15] on the convergence of fictitious play in two-person zero-sum games, we obtain the following final result:

Theorem 3. *The scalarized compound game $\overline{\Gamma}_{sc}$ defined by* (8) *has the fictitious play property.*

Theorem 3 induces the following procedure to compute multi-criteria security strategies according to definition 1:

Algorithm to compute MCSS: Given a two-player MOG Γ with k payoffs $u_1^{(1)}, \ldots, u_1^{(k)}$ for player 1 (and possibly unknown payoffs for player 2), we obtain a MCSS along the following steps:

1. Assign strictly positive weights $\alpha_{01}, \ldots, \alpha_{0k}$, satisfying $\sum_{i=1}^{k} \alpha_{0i} = 1$, to each goal, and set up the scalarized auxiliary compound game $\overline{\Gamma}_{sc}$ by virtue of expressions (8), (6) and (7).
   Observe that, as we can choose the weights arbitrarily, these give us a method to *prioritize* different goals. However, practical experiments indicated that different choices of priorities (α-values) have only a minor if not negligible effect on the particular result of the computation.
2. Run fictitious play in $\overline{\Gamma}_{sc}$, stopping when the desirable precision of the equilibrium approximation is reached. In our experiments, we stopped when the difference between the intermediate result vectors θ^*_{t-1} and θ^*_t at steps t and $t-1$ has become less than an adjustable threshold $\delta > 0$ in the 1-norm.

3. The result vector θ^* is directly the sought multi-criteria security strategy, whose assurances are given by the respective expected payoffs of the opponents. In case of matrix games, where the i-th payoff is given by a matrix $\boldsymbol{A}_i$, the sought assurances are $v_i = (\theta^*)^T \boldsymbol{A}_i \phi_i^*$ for $i = 1, 2, \ldots, k$, where $\phi_1^*, \ldots, \phi_k^*$ are the other player's equilibrium strategy approximations obtained along the fictitious play.

5 Experimental Evaluation

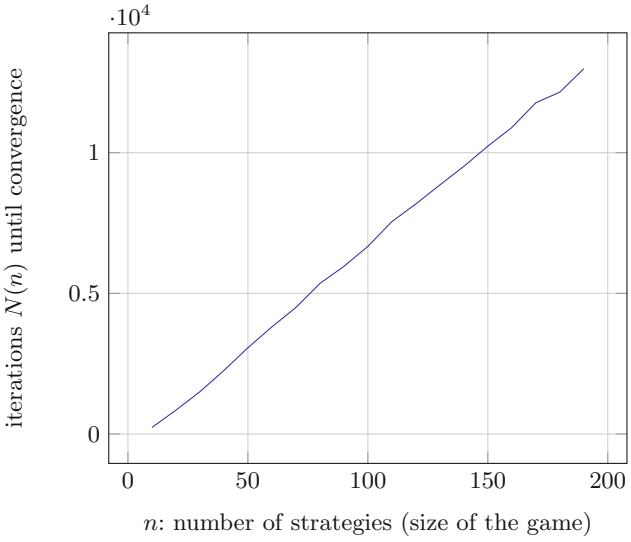
We stress that theorem 3 asserts the fictitious play property for the games constructed, yet does not limit numerical solution techniques to a particular algorithm (not even to fictitious play). Our experimental implementation used the basic (and non-optimized) fictitious play procedure [15], but can easily be replaced by more sophisticated algorithms (e.g., [20]) to gain speed. Our tests were done on a 3 x AMD Opteron 6212 machine, having 2.6 GHz 24 cores (virtualized), 96 GB RAM, and 1 TB disk space.

Towards a (non-application-specific) performance evaluation, we created random payoff matrices to simulate arbitrary matrix game structures (matrices with independent and uniformly distributed Bernoulli random entries) ranging from 2 to 170 strategies (in steps of 2) for the honest player, seeking to secure its behavior in terms of two security goals. In each setting, we ran (at least) 50 trials, taking the average number of iterations until convergence as the empirical performance indicator. Convergence is said to be reached once the change in the payoff-values $v_1, \ldots, v_k$ (per security goal) between two iterations has become less than a threshold $\delta = 0.01$ in the 1-norm.[2] Figure 1a plots the results.

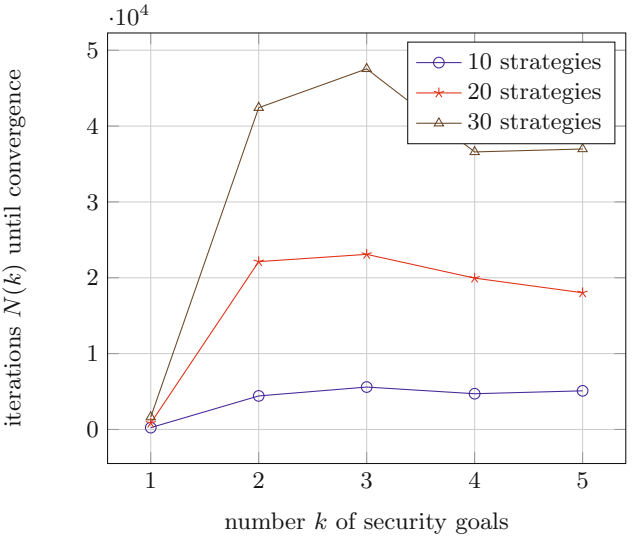
Fictitious play has shown to be numerically stable, yet suffers from slow convergence (without optimizations) and memory shortage in case of games with many goals (each of which corresponds to a player with its own payoff structure). In the latter cases, the computation may be parallelized towards a speed-up by assigning each player its own processor and memory. The temporal speed under parallelization is then mostly determined by the communication overhead, which in a multi-processor CPU is not too much of a problem.

As expected, the maximal number of iterations grows with the size of the strategy sets and the number of security goals. Towards an empirical estimate of asymptotic complexity in terms of the game's *size* (number of strategies), we fitted a linear model to the plot of $N(n)$ (Figure 1a). Here, n is the number n of strategies, and the model took the form $N = a \cdot n + b + \varepsilon$ with an error term ε being normally distributed. The parameter estimates came up to $a \approx 71.5657, b \approx -507.7625$. The normality hypothesis on the residual term ε was accepted by a Shapiro-Wilks test with a p-value of ≈ 0.8918 at a confidence level of $\alpha = 0.95$. Hence, we may – on empirical evidence – assume a growth of the iteration count N that is proportional to the number n of strategies, giving

[2] Notice that convergence in the fictitious play process as defined above implies convergence under our modified criterion by the continuity of the payoff functions.



(a) Convergence, depending on the strategy set sizes



(b) Convergence, depending on the number of security goals

Fig. 1. Complexity of computing two-criteria security strategies

linear asymptotic average-case complexity $N \in O(n)$. The same linear relationship was also confirmed for trials in 3 and 4 dimensions (using smaller games in terms of strategy counts, though). Interestingly, the constants within the big-O were roughly equal between 2, 3 and 4 dimensions, indicating that convergence rates are only mildly affected by the number of security goals (dimensions).

This is somewhat confirmed by the plot in Figure 1b, although a more thorough empirical investigation needs to be done. A deeper exploration of both observations will be done with games that correspond to network security protocols (see the related work section 6), and will appear in companion work to this.

The convergence speed (number of iterations) is rather slow: the computation took about 15 minutes computing time until a precision of $\delta = 0.01$, and another 15 minutes to undercut $\delta = 0.001$ in three dimensions with 100 strategies. Figure 2 shows the evolution of the difference between adjacent equilibrium profile approximations (beliefs) over the iterations of a single run, taking 200 strategies in two dimensions until a precision of $\delta = 0.001$ is reached. As the figure shows, the algorithm quickly approaches the equilibrium, but slows down substantially near the optimum. So, although we get a quick-and-dirty first approximation, retrieving more accurate results upon fictitious play takes some time. Section 6.2 describes an application to network security, based on multipath transmission.

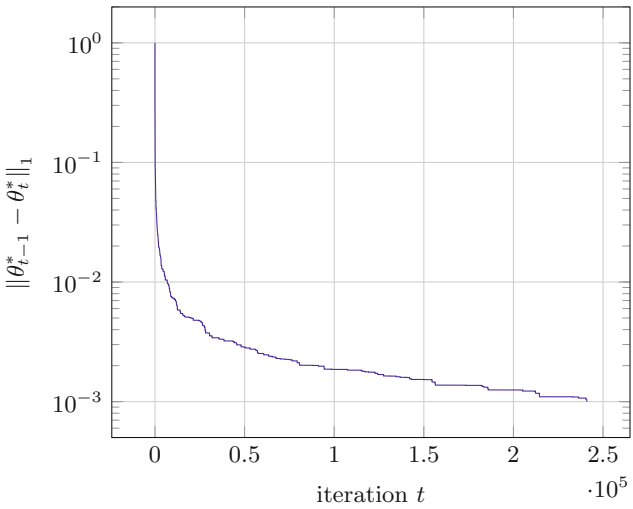


Fig. 2. Convergence speed plot

The speed of convergence of fictitious play in general games is known to be very slow, as was demonstrated by [4] on a concrete example game, where the FP process takes exponentially many rounds until the equilibrium is reached. Alternatively, convergence may be measured by considering the difference in the *payoffs*, rather than the behavior profiles (beliefs), such as we did in our experimental implementation. These may converge even though the distributions themselves may oscillate.

However, the slow convergence of regular fictitious play may – in large games – become unhandy, thus calling for replacements by more refined and sophisticated learning techniques. Inspecting the applicability of such alternatives is an interesting direction of future research.

6 Related Work

The idea of Pareto-optimal security strategies (POSS) is not new and has previously been introduced in [7,6,18]. This prior work appears as a special case of definition 1 when the games are finite. Infinity of action spaces, which arise when continuous parameters (such as timing) were not covered by this preliminary work. Treating communication as a game is a well-researched field, with a comprehensive account given by [2], and much precursor work (such as [21]). Game-theory has in the past as well been used to negotiate optimal service and operational level agreements (see [11,8] among others) and to quantitatively analyze security in ad hoc networks [22] under several optimality concepts (among which is Pareto-optimality). Our work aids and further substantiates this direction of research. An interesting yet unexplored relation to our work also exists in the results of [16], who consider a "non-static" gameplay. This direction is one of future considerations.

6.1 Multipath Transmission

A fruitful application is a game-theoretic model of multipath transmission. Roughly speaking, the game is about an honest sender attempting to communicate over a network that is partially under the attacker's control. The attacker is not constrained in its computational power, but limited to control a fixed maximal number of nodes, by which it can read and insert network traffic at its own will. The honest player's goal is to deliver a message to a designated receiver, while the payload remaining *confidential* and *authentic*, and with the maximum probability of delivery (*availability*). The gameplay is by the honest party (player 1) randomly choosing transmission paths, while the attacker (player 2) randomly chooses nodes to sniff, which – in its simplest form just described – makes the scenario almost a diagonal game. An illustration is given in Figure 3.

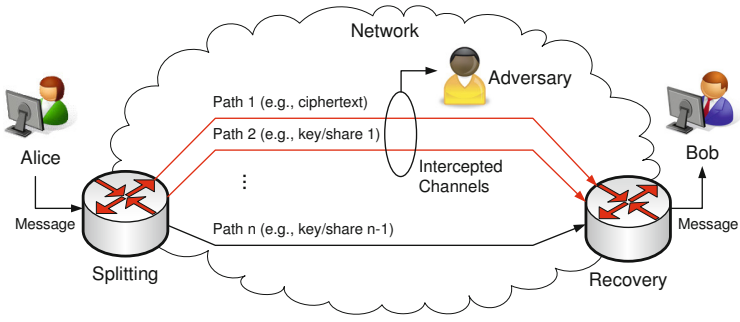


Fig. 3. Illustration of multipath transmission

We leave the protocol-, game- and cryptography-details aside here (referring the interested reader to [5,19,3,12,14] to fill these gaps), and confine ourselves

to stating that experimental evaluations on real life enterprise network topologies lead to small games (after eliminating redundant and dominated strategies) that are easy to handle. This is mostly due to low connectivity (many enterprise network *backbones* have a graph vertex connectivity of two, for reasons of redundancy). Realistic wide area topologies would follow an Erdös-Rényi or scale-free topology, which we simulate in the course of a research project (see the acknowledgement) on which we will report in subsequent work. Here, for the sake of generality, this example shall merely substantiate the applicability of the theoretical concept of Pareto-optimal security strategies, while our evaluation will be on matrix games with randomly chosen payoff structures.

6.2 Example: Security of Multipath Transmission

Nevertheless, the method appears viable to compute quantitative security of multipath transmission on a given network topology. As an example, consider the network topology depicted in Figure 4, where Alice wishes to securely send a message to Bob over the network. Hereby, a message m is called *secure*, if its transmission is confidential, the payload is authentic and the delivery does not fail (availability). Hence, we have three goals, i.e., three dimensions.

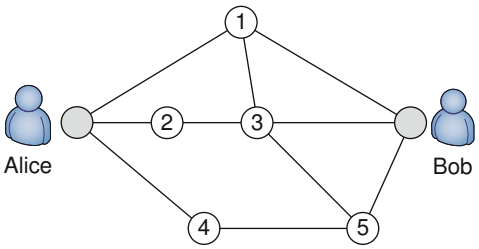


Fig. 4. Example network

The transmission protocol uses two paths and a one-time pad encryption, sending the key k over one path, and the ciphertext $c = m \oplus k$ over the second path, where $\oplus$ denotes the bitwise XOR (note that this scheme is trivial to generalize to the usage of $n > 2$ paths).

The adversary is allowed to conquer any two nodes between Alice and Bob (excluding the two, for obvious reasons), and is computationally unbounded (i.e., we are after unconditional security here).

The game's payoff structure is composed from three indicator functions of success, measuring confidentiality as $u_1^{(\text{conf})} = 1 :\iff$ [the attacker misses either k or c], availability as $u_1^{(\text{avail})} = 1 :\iff$ [the attacker fails to intercept k or c], and authenticity. This is achieved by the protocol in [14], and yields $u_1^{(\text{auth})} = 1 :\iff$ [the attacker fails to conquer at least one of the chosen paths]. The strategy set for player 1 is the set of pairs of disjoint transmission paths (a total of $|PS_1| = 3$ strategies). The strategy set for player 2 is the set of two-element subsets of $\{1, 2, 3, 4, 5\}$, giving a total of $|PS_2| = \binom{5}{2} = 10$ strategies. The

payoff for player 1 is the vector $\boldsymbol{u}_1 = (u_1^{(\mathrm{conf})}, u_1^{(\mathrm{avail})}, u_1^{(\mathrm{auth})})$. The importance weights are $(\alpha_{0,\mathrm{conf}}, \alpha_{0,\mathrm{avail}}, \alpha_{0,\mathrm{auth}}) = (1/3, 1/3, 1/3)$.

The fictitious play process converged within 6 iterations until an accuracy of $\delta < 10^{-3}$, giving the final multicriteria security strategy $\theta^* = (\frac{1}{3}, \frac{1}{3}, \frac{1}{3})$, with assurance $\boldsymbol{v} = (\frac{2}{3}, 0, \frac{2}{3})$. This is indeed what we expect, since if the attacker intercepts one of the paths, the message remains confidential and cannot be forged unnoticeably ($u_1^{(\mathrm{conf})} = 1 = u_1^{(\mathrm{auth})}$), but it can become destroyed ($u_1^{(\mathrm{avail})} = 0$). The assurance vectors thus give the conditional probability $\Pr[m \textit{ is authentic and has not been disclosed} \mid m \textit{ was correctly delivered}] \geq 2/3$, but the unconditional likelihood $\Pr[\text{delivery of } m \text{ can be disrupted}] = 1$. By the properties of MCSS, this is the best that the attacker can do. The protocol is as such insecure, as it is vulnerable to denial-of-service, although it can be made arbitrarily and unconditionally secure against eavesdropping (under the given adversary model) by repeating the process on a sequence of packets $m_1, m_2, \ldots, m_\ell$ whose bitweise XOR recovers $m = m_1 \oplus m_2 \oplus \cdots \oplus m_\ell$. Then, the likelihood to disclose m is $2^{-O(\ell)}$, if all ℓ messages are delivered according to the security strategy θ^*.

It is straightforward to apply the technique to other more efficient protocols like [19,5], and to take further probabilistic security in the network into account, by replacing the payoff functions accordingly.

7 Conclusion

Fictitious play has been demonstrated as a working method to numerically compute security strategies towards playing safe in multiple regards (security goals). The axiomatic characterization of multi-criteria security strategies as Pareto-Nash equilibria, which in turn can be computed as Nash-equilibria of multi-player games, induces a sequence of simple and straightforward transformations that culminate in a game enjoying the fictitious play property. In addition, we gain a degree of freedom to assign importance weights to different security goals, although these seem to have only minor (if not negligible) influence on the actual outcome (equilibrium) that is computed. Nevertheless, it adds an interesting aspect to practical applications by showing that a "prioritization" between security goals is not necessarily useful in general.

Aspects of future work are non-static game-plays, improved variants of fictitious play and examining complexities to more detail. As a showcase application, we will apply our algorithms to problems of establishing confidential, authentic and reliable communication in large scale computer networks by means of multipath transmission. Given the available cryptographic fundament, quantifying security in terms of Pareto-optimal security strategies then boils down to a straightforward application of our numerical method presented here.

Acknowledgment. This work was supported by the Austrian Research Promotion Agency (FFG) under project grants no. 836287. Furthermore, we thank the anonymous reviewers for valuable suggestions and for drawing our attention to interesting aspects of future research.

References

1. Acosta Ortega, F., Rafels, C.: Security strategies and equilibria in multiobjective matrix games. Working Papers in Economics 128, Universitat de Barcelona. Espai de Recerca en Economia (2005), http://ideas.repec.org/p/bar/bedcje/2005128.html
2. Alpcan, T., Başar, T.: Network Security: A Decision and Game Theoretic Approach. Cambridge University Press (2010)
3. Ashwin Kumar, M., Goundan, P.R., Srinathan, K., Pandu Rangan, C.: On perfectly secure communication over arbitrary networks. In: PODC 2002: Proceedings of the Twenty-First Annual Symposium on Principles of Distributed Computing, pp. 193–202. ACM, New York (2002)
4. Brandt, F., Fischer, F., Harrenstein, P.: On the rate of convergence of fictitious play. In: Kontogiannis, S., Koutsoupias, E., Spirakis, P.G. (eds.) SAGT 2010. LNCS, vol. 6386, pp. 102–113. Springer, Heidelberg (2010)
5. Fitzi, M., Franklin, M.K., Garay, J.A., Vardhan, S.H.: Towards optimal and efficient perfectly secure message transmission. In: Vadhan, S.P. (ed.) TCC 2007. LNCS, vol. 4392, pp. 311–322. Springer, Heidelberg (2007)
6. Ghose, D.: A necessary and sufficient condition for pareto-optimal security strategies in multicriteria matrix games. Journal of Optimization Theory and Applications 68(3), 463–481 (1991)
7. Ghose, D., Prasad, U.R.: Solution concepts in two-person multicriteria games. Journal of Optimization Theory and Applications 63(2), 167–189 (1989)
8. Kaminski, H., Perry, M.: A framework for automatic SLA creation. Tech. rep. The University of Western Ontario, Computer Science Publications (2008)
9. Lozovanu, D., Solomon, D., Zelikovsky, A.: Multiobjective games and determining pareto-nash equilibria. Buletinul Academiei de Stiinte a Republicii Moldova Matematica 3(49), 115–122 (2005), ISSN 1024-7696
10. McKelvey, R.D., McLennan, A.M., Turocy, T.L.: Gambit: Software tools for game theory, version 0.2007.12.04 (2007), http://gambit.sourceforge.net
11. Moroni, S., Figueroa, N., Jofre, A., Sahai, A., Chen, Y., Iyer, S.: A game-theoretic framework for creating optimal SLA/contract. Tech. Rep. HPL-2007-126, HP Laboratories Palo Alto (2007)
12. Rass, S., Schartner, P.: A unified framework for the analysis of availability, reliability and security, with applications to quantum networks. IEEE Transactions on Systems, Man, and Cybernetics – Part C: Applications and Reviews 41(1), 107–119 (2011)
13. Rass, S.: On game-theoretic network security provisioning. Springer Journal of Network and Systems Management 21(1), 47–64 (2013)
14. Rass, S., Schartner, P.: Multipath authentication without shared secrets and with applications in quantum networks. In: Proceedings of the International Conference on Security and Management (SAM), July 12-15, vol. 1, pp. 111–115. CSREA Press (2010)
15. Robinson, J.: An iterative method for solving a game. Annals of Mathematics 54, 296–301 (1951)
16. Ryu, C., Sharman, R., Rao, H., Upadhyaya, S.: Security protection design for deception and real system regimes: A model and analysis. European Journal of Operational Research 201(2), 545–556 (2010), http://www.sciencedirect.com/science/article/B6VCT-4VXTSK1-2/2/9ffe61e9aa467ce2271adfa338f27842

17. Sela, A.: Fictitious play in 'one-against-all' multi-player games. Economic Theory 14, 635–651 (1999), http://dx.doi.org/10.1007/s001990050345
18. Voorneveld, M.: Pareto-optimal security strategies as minimax strategies of a standard matrix game. Journal of Optimization Theory and Applications 102(1), 203–210 (1999)
19. Wang, Y., Desmedt, Y.: Perfectly secure message transmission revisited. IEEE Transactions on Information Theory 54(6), 2582–2595 (2008)
20. Washburn, A.: A new kind of fictitious play. Tech. rep., Operations Research Department, Naval Postgraduate School, Monterey, California 93943, copyright by John Wiley & Sons, Inc. (2001)
21. Ying, Z., Hanping, H., Wenxuan, G.: Network security transmission based on bimatrix game theory. Wuhan University Journal of Natural Sciences 11(3), 617–620 (2006)
22. Yu, W., Liu, K.J.R.: Game theoretic analysis of cooperation stimulation and security in autonomous mobile ad hoc networks. IEEE Transactions on Mobile Computing 6(5), 507–521 (2007)