How to Safely Exploit Predictions in General-Sum Normal Form Games

Abstract

Given a prediction of opponent behavior in a general-sum two-player normal form game, it is difficult to select a strategy that balances the opportunity to use the prediction to inform one’s action with the risk of becoming predictable. We propose Restricted Stackelberg Response with Safety (RSRS), a novel way of generating such a strategy. RSRS uses an $r$-safe Stackelberg equilibrium in a modified game, which is created to reflect the assumption that the prediction might be inaccurate. With appropriate parameter selection, RSRS produces strategies that can play well against the prediction, respond well against a best-responding opponent, or guard against worst-case outcomes. We describe an algorithm that selects appropriate parameter values, which we have tested on multiple general-sum games, comparing its performance to that of other algorithms.

Introduction

How should an agent respond when given a prediction of opponent behavior in a general-sum two-player normal form game? Selecting the strategy with the highest payoff against the prediction provides optimal performance if the prediction is correct, but can be arbitrarily bad if the prediction is incorrect. Playing a maximin strategy guarantees a payoff equal to the safety value of the game, but at the cost of performance against the prediction. Playing a Nash equilibrium is equivalent to assuming the opponent will play the Nash equilibrium. Both options ignore the value of the prediction.

We introduce Restricted Stackelberg Response with Safety (RSRS), a novel method of choosing a mixed strategy given a general-sum game and a prediction of the opponent’s strategy. RSRS uses a prediction weight parameter, $u$, to determine how much to guard against a best-responding opponent, and a risk factor parameter, $r$, to determine how much to guard against a worst-case outcome.

RSRS is not a prediction method, it provides a way to use a prediction made by an external algorithm to produce a strategy that reflects a controlled tradeoff between best-responding to the prediction and dealing with a best-responding opponent, while also providing a guarantee of worst-case performance. We use fictitious play for prediction, but RSRS can be used with any prediction method that produces a probability distribution over opponent moves.

The main contributions of this paper are the Restricted Stackelberg Response with Safety (RSRS) method, which is applicable to general sum games, a method to find appropriate weight values for RSRS, and experimental results in a demonstration game. We also prove the uniqueness of Restricted Nash Response (RNR) in zero-sum games.

Terminology. A game $G$ consists of a set of players $\{1, 2\}$, a set of actions for each player $M_1 = \{m_1^1 \ldots m_1^n\}, M_2 = \{m_2^1 \ldots m_2^n\}$, and a set of utility functions $U_1 : M_1 \times M_2 \rightarrow \mathbb{R}, U_2 : M_1 \times M_2 \rightarrow \mathbb{R}$. In a zero-sum game $s \in \Delta(M_1 \times M_2)$, where $s_1 \in \Delta M_1, s_2 \in \Delta M_2$ are the strategies adopted by player 1 and player 2 respectively.

We abuse notation to define $U_i(s_1, s_2) = \mathbb{E}_{m_1 \sim s_1, m_2 \sim s_2}[U_i(m_1, m_2)]$ as the expected outcome for player $i$ when actions are drawn from the distributions $s_1$ and $s_2$. The Nash equilibrium of a game is a set of strategies $S_1, S_2$ such that $U_1(S_1, S_2) \geq \max_{m_1 \in M_1} U_1(m_1, S_2)$ and $U_2(S_1, S_2) \geq \max_{m_2 \in M_2} U_2(S_1, m_2)$. The safety value of the opponent is $V^*_i = \min_{m_2 \in \Delta M_2} \max_{m_1 \in \Delta M_1} U_i(m_1, m_2)$. This is the greatest amount player $i$ can guarantee for itself regardless of the opponent’s action. Note that for general-sum games, this value may be lower than the expected payoff of any Nash equilibrium of the game.

If player 1 is designated a Stackelberg leader (Fudenberg and Tirole 1991) for the game, they select a mixed strategy which is observed by player 2 before player 2 selects their strategy. The Stackelberg equilibrium of a game is a set of strategies $S_1, S_2$ such that $S_1 = \arg\max_{s_1 \in \Delta M_1} U_1(s, \arg\max_{s_2 \in \Delta M_2} U_2(s, s'))$ and $S_2 = \arg\max_{s_2 \in \Delta M_2} U_2(S_1, s)$.

Demonstration Game. We have examined the performance of our method across a wide variety of games. The advantages of RSRS can most easily be observed in competitive general-sum games, which allow players to have some common interest. The game we will use to show the properties of our RSRS method is Rock/Spock/Paper/Lizard/Scissors – a variant of Rock/Paper/Scissors with 5 moves (Table 1). Each action beats two other actions, and is beaten in turn by the two re-
maintaining actions. Players receive a payoff of 1 for a win, -1 for a loss, and 0 for a tie. In addition, both players receive .5 when adjacent moves are played and lose .5 when non-adjacent moves are played. The game has a unique Nash equilibrium. In this game the best response to a move is distinct from the worst-case outcome, which allows us to distinguish between guarding against a best-responding opponent and guarding against any bad outcome.

<table>
<thead>
<tr>
<th></th>
<th>Rock</th>
<th>Spock</th>
<th>Paper</th>
<th>Lizard</th>
<th>Scissors</th>
</tr>
</thead>
<tbody>
<tr>
<td>Rock</td>
<td>0.0</td>
<td>-5.1,5</td>
<td>-5.1,5</td>
<td>-5.1,5</td>
<td>1.5,-.5</td>
</tr>
<tr>
<td>Spock</td>
<td>1.5,-.5</td>
<td>0.0</td>
<td>-5.1,5</td>
<td>-5.1,5</td>
<td>1.5,-.5</td>
</tr>
<tr>
<td>Paper</td>
<td>.5,-1.5</td>
<td>1.5,-.5</td>
<td>0.0</td>
<td>-5.1,5</td>
<td>-1.5,.5</td>
</tr>
<tr>
<td>Lizard</td>
<td>-1.5,.5</td>
<td>.5,-1.5</td>
<td>1.5,-.5</td>
<td>0.0</td>
<td>-5.1,5</td>
</tr>
<tr>
<td>Scissors</td>
<td>-5.1,5</td>
<td>-1.5,.5</td>
<td>.5,-1.5</td>
<td>1.5,-.5</td>
<td>0.0</td>
</tr>
</tbody>
</table>

Table 1: Payoffs for Rock/Spock/Paper/Lizard/Scissors.

**Related Work**

A lot of work has been done to develop models of opponents in multi-agent environments. The simplest opponent model, a stationary distribution over actions, is the model underlying fictitious play (Fudenberg and Levine 1998), where agents assume a stationary opponent, calculate the most likely distribution for such an opponent, and best-respond.

**AWESOME** (Conitzer and Sandholm 2007) best-responds to a stationary opponent, but arrives at a Nash Equilibrium in self-play. It observes the opponent’s play, detects stationarity, exploits it if possible, and retreats to equilibrium play when it detects a non-stationary opponent. Competitive safety analysis (Tennenholtz 2002) provides performance guarantees relative to the Nash equilibrium. RWYWE (Ganzfried and Sandholm 2015) is guaranteed to achieve the safety value of the game immediately, not just as the long-term average. PCM(A) (Powers, Shoham, and Vu 2007) achieves a best response against stationary opponents and opponents who converge to stationary, achieves a pareto-optimal Nash equilibrium in self-play, and is guaranteed to achieve its security value, regardless of the opponent. CMLeS (Chakraborty 2014) guarantees safety, and optimality when playing against adaptive opponents with some memory limit. (Wang et al. 2011) assume local stationarity in the opponent strategy and learn best response, learning a new one when a change in the opponent strategy is detected. In (Elidrisi and Gini 2012) an agent predicts the next opponent action and reasons about the prediction accuracy to adapt the prediction. Similarly, (Hernandez-Leal, Munoz de Cote, and Sucar 2014) learns an MDP model of the opponent strategy, while detecting strategy switches.

Adaptive dynamic learning (Burkov and Chaib-draa 2007) learns how agents respond to sequences of play. This shows that an opponent doesn’t need to know an agent’s exact model to exploit it. An exploiting agent capable of taking advantage of a bounded memory opponent is presented in (de Cote and Jennings 2010). Instead of playing a move with the highest expected payoff, it calculates a sequence of actions with the highest expected payoff. This highlights the difficulty of detecting an exploiting opponent, as the opponent may not be playing an immediate best response.

One problem we had to address when using the Stackelberg equilibrium is an opponent that may not be strictly best-responding. We deal with this by using an exponential response function to model an opponent that is biased towards best-responding. An alternative approach is in (Pita et al. 2010), which describes a modification of the Stackelberg equilibrium to handle bounded-rational opponents with human biases in probability estimation.

The approaches most similar to our work are Safe Policy Selection (SPS) (McCracken and Bowling 2004) and Restricted Nash Response (RNR) (Johanson, Zinkevich, and Bowling 2007), which we describe next.

**Restricted Nash Response (RNR)**

RNR was presented in (Johanson, Zinkevich, and Bowling 2007) as a method of exploiting a prediction of opponent behavior in poker. It finds a strategy by constructing a modified game, and taking the Nash equilibrium of that game.

To calculate a RNR for a zero-sum game $G$, a prediction $s_2 \in \Delta M_2$ and a weight $w \in [0,1]$ are used to construct a modified game $G'$ with $M_1' = M_1$, $M_2' = M_2$, $U_1' = U_1$, and $U_2'(m_1,m_2) = w \times U_1(m_1,s_2) + (1-w) \times U_1(m_1,m_2)$.

Figure 1 shows the performance of RNR with different $w$ values in Rock/Spock/Paper/Lizard/Scissors. The prediction is Rock and performance is measured against the prediction, against a best-response, and in the worst case. When $w = 1$, RNR will play a best response to the prediction. When $w = 0$, RNR will play a Nash equilibrium of the original game. Intermediate values will cause the generated strategy to change discontinuously between those two extremes. In general-sum games, increasing the weight value can reduce performance against the prediction. RNR calculates a strategy by computing the Nash equilibrium of a modified game. We will show that the general-sum modified game created to calculate a RNR has a unique Nash equilibrium whenever the original game has a unique Nash equilibrium.

Let $G'$ be the modified game created from $G$ to calculate RNR, with utility function $U'$ where $U_1'(m_1,m_2) = w \times U(m_1,m_2) + (1-w) \times U(m_1,m_2)$ and $U_2' = -U$. Assume $G'$ has two distinct Nash equilibria $s$ and $s'$. Two equilibria are distinct if each player strictly prefers to play their equilibrium strategy in their equilibrium: $U_1(s_1,s_2 > U_1(s_1',s_2)$, $U_1(s_1',s_2) > U_1(s_1,s_2)$, $U_2(s_1,s_2) > U_2(s_1',s_2)$, and $U_2(s_1',s_2) > U_2(s_1,s_2')$. If the preference is weak then the two equilibria are part of the same connected component and it doesn’t present the same dilemma for RNR because the players can play either strategy and achieve the same payoff.

Construct a new game $G''$ from $G'$ with moves $s_1, s_1' \in \Delta M_1''$ and $s_2, s_2' \in \Delta M_2''$ and payoffs equal to playing the corresponding strategies in $G''$.

Because $s$ and $s'$ are distinct equilibria, we have:

$$w \times U(s_1, p) + (1-w) \times U(s_1, s_2) > w \times U(s'_1, p) + (1-w) \times U(s'_1, s_2)$$

(1)
within method of responding to a prediction of opponent behavior equilibria in the modified game created for RNR. SPS is a method of selecting by gradually increasing the forming well against predictable opponents. It does this because the modified game has a unique equilibrium. Spock/Paper/Lizard/Scissors. Calculation of RNR is possible, so there cannot be two distinct Nash equilibria in the modified game created for RNR.

**Safe Policy Selection (SPS)**

SPS was presented in (McCcracken and Bowling 2004) as a method of responding to a prediction of opponent behavior in Rock/Paper/Scissors. Given a game \( G \) with safety value, \( V_G \) an \( r \)-safe strategy \( s_1^r \) is one whose worst case payoff is within \( r \) of the safety value \( \min_{s_2 \in \Delta M_2} U(s_1^r, s_2) \geq V_G - r. \) SPS is a method of selecting \( r \) values to guarantee that the player receives the safety value of the game while also performing well against predictable opponents. It does this by gradually increasing the \( r \) value, but reducing it whenever the expected payoff of the agent’s strategy against the opponents move is less than the safety value of the game, \( r_n = r_{n-1} + 1/n + U_1(s_1^{n-1}, m_2^{n-1}) - V_G. \)

\[
w \times U(s_1', p) + (1 - w) \times U(s_1', s_2') > w \times U(s_1, p) + (1 - w) \times U(s_1, s_2') \quad (2)
\]
\[
- U(s_1', s_2') > - U(s_1', s_2) \quad (3)
\]
\[
- U(s_1, s_2) > - U(s_1, s_2') \quad (4)
\]

From 1 and 3 we get:
\[
w \times U(s_1, p) + (1 - w) \times U(s_1, s_2) > w \times U(s_1', p) + (1 - w) \times U(s_1', s_2') \quad (5)
\]

From 5 and 2 we get:
\[
w \times U(s_1, p) + (1 - w) \times U(s_1, s_2) > w \times U(s_1', p) + (1 - w) \times U(s_1', s_2') \quad (6)
\]

From 6 and 4 we get:
\[
w \times U(s_1, p) + (1 - w) \times U(s_1, s_2) > w \times U(s_1, p) + (1 - w) \times U(s_1, s_2) \quad (7)
\]

This is not possible, so there cannot be two distinct Nash equilibria in the modified game created for RNR.

**RSR with Safety (RSRS)**

The RSRS for player 1 in game \( G \) with prediction \( p \in \Delta M_2 \), prediction weight \( w \in [0, 1] \), and risk factor \( r \in \mathbb{R} \) is the mixed strategy for player 1 that maximizes its expected payoff given the assumption that, with probability \( w \), the opponent will play according to the prediction \( s_2 \in \Delta M_2 \) and, with probability \( 1 - w \), it will best-respond, subject to the constraint that its expected payoff when played against any opponent action is at least \( V_G^r - r \).

To compute RSRS we construct a new payoff function \( U_1' \), which reflects the assumption that the opponent will play according to the prediction with probability \( w \):
\[
U_1'(m_1, m_2) = w \times U_1(m_1, p) + (1 - w) \times U_1(m_1, m_2)
\]

The RSRS for player 1 is the probability distribution \( s_1 \in \Delta M_1 \), which maximizes the expected value of \( U_1' \) under the assumption that player 2 will best respond to the mixed strategy of player 1, subject to the constraint that \( U_1(s_1, m_2) \geq V_G^r - r \) for all \( m_2 \in M_2 \). Assuming that the opponent is best-responding to the action of player 1 is equivalent to assuming that player 1 is a Stackelberg leader in the game. Playing the Nash equilibrium in this situation forfeits the potential benefit that comes from being a Stackelberg leader.

To compute the optimal strategy for player 1 we use a modification of the technique outlined in (Conitzer and
Sandholm 2006). For each opponent action $m_2 \in M_2$ we find (if possible) a mixed strategy $s_1 \in \Delta M_1$ which maximizes the agent’s payoff in the modified game when played against $m_2$, for which $m_2$ is a best-response to $s_1$ for player 2, subject to $U_1(s_1, m_2) \geq V_G^1 - r$ for all $m_2 \in M_2$.

For each $m_2 \in M_2$ maximize over $s_1 \in \Delta M_1$:

$$s_1 = \arg \max_{s_1 \in \Delta M_1} U_1^*(s_1, m_2)$$

subject to

$$\forall m_2 \in M_2, U_2(s_1, m_2) \geq U_2(s_1, m'_2)$$

and

$$\forall m_2 \in M_2, U_1(s_1, m_2) \geq V_G^1 - r$$

Solving a set of equations for each opponent action will give us at least 1 and up to $n$ mixed strategies for player 1. The mixed strategy $s_1$ with the highest expected value for $U_1^*$ against the opponent’s best response is the RSRS.

The values chosen for probability weight ($w$) and risk factor ($r$) control the tradeoff between performance against the prediction, performance against a best-responding opponent ($w$), and performance against a worst-case opponent ($r$).

For a fixed risk factor there are many probability weights that produce the same strategy – changes in $w$ either produce no change or a discontinuous jump to a completely new strategy. In contrast, changes to $r$ produce a continuous variation between a minimax strategy and a prediction exploiting strategy. Note that the effect of $r$ dominates the effect of $w$. If $r = 0$, the $w$ value has no effect. A value for $r$ can be selected according to SPS. We select a value for $w$ by calculating the relative posterior probability of the opponent following the prediction and the opponent best-responding.

Figure 3 shows the effects of $w$ on the performance of RSRS playing Rock/Spock/Paper/Lizard/Scissors with a prediction of Rock. Performance is measured against the prediction, against a best-responding opponent, and in the worst case. As $w$ varies the resulting strategy changes discontinuously. At $w = 0$ it produces a Stackelberg equilibrium, at $w = 1$ it produces a best response to the prediction. As $w$ increases, performance against the prediction strictly increases, unlike RNR. In contrast, $r$ produces a continuous variation of performance similar to Figure 2, ranging from the minimax strategy ($r = 0$) to a best response ($r = 1.5$).

We can characterize the change in performance produced by a change in $w$ for a fixed $r$ in terms of the trade-off between performance against the prediction and performance against a best-responding agent. For example, in Rock/Spock/Paper/Lizard/Scissors, with a prediction of Rock and a weight of 1.5, RSRS produces $(0, .6, 0, 0, .4)$ when $w < .6$ and $(0, 1, 0, 0, 0)$ when $w \geq .6$. The first equilibrium produces a payoff of .7 when played against the prediction or a best-responding opponent. The second equation produces a payoff of 1.5 when played against the prediction, and a payoff of $-.5$ when played against a best-responding opponent.

Note that when $w$ transitions from below .6 to above .6, it gains .8 in expected payoff against the prediction and loses 1.2 against a best-responding opponent. The expected gain against the prediction is 2/3 as much as the expected loss against a best-responding opponent. This matches the relative probability of those two events expressed by a $w$ value of .6. This is not a coincidence, as we shall prove below.

We are interested in values of $w$ between two regions in each of which RSRS for a fixed $r$ does not change. For those $w$ values we denote with $rsrs_{w+}$ and $rsrs_{w-}$ respectively the RSRS for the region with weight values higher or lower than $w$. We denote with $br_{w+}$ and $br_{w-}$ the best responses to those strategies. Assume we are given a game $G$, a prediction $p \in \Delta M_2$, and a $w$ where the RSRS changes. If there is a $\delta$ such that for all $0 < \epsilon < \delta$ the RSRS with weight $w + \epsilon$ is the same for all $\epsilon$, and the RSRS with weight $w - \epsilon$ is the same for all $\epsilon$, and $rsrs_{w+} \neq rsrs_{w-}$, we will prove:

**Lemma 1.** Reducing the value of $w$ used to compute a RSRS can only improve performance against a best-responding opponent, i.e. $U_1(rsrs_{w+}, br_{w+}) < U_1(rsrs_{w-}, br_{w-})$

**Proof.** Consider $U_1(rsrs_{w+}, p) - U_1(rsrs_{w-}, p)$ and $U_1(rsrs_{w+}, br_{w+}) - U_1(rsrs_{w-}, br_{w-})$, which represent the relative performance of $rsrs_{w+}$ compared to $rsrs_{w-}$ against the prediction and a best-responding opponent respectively. If both are positive or both are negative then $rsrs_{w+}$ or $rsrs_{w-}$ would be strictly superior to the other, which contradicts that they were generated as payoff-maximizing distributions. Because $rsrs_{w+}$ was found by maximizing performance in $U_{w+}^+$ against a best-responding opponent, we know that $U_{w+}^+(rsrs_{w+}, br_{w+}) > U_{w+}^+(rsrs_{w-}, br_{w-})$. From the definition of $U_{w+}^+$ this gives us

$$(w + \epsilon)U_1(rsrs_{w+}, p) + (1 - w - \epsilon)U_1(rsrs_{w+}, br_{w+}) > (w + \epsilon)U_1(rsrs_{w-}, p) + (1 - w - \epsilon)U_1(rsrs_{w-}, br_{w-})$$

$$+ (1 - w - \epsilon)U_1(rsrs_{w+}, br_{w+})$$

Similarly, for $rsrs_{w-}$ we have

$$(w - \epsilon)U_1(rsrs_{w-}, p) + (1 - w + \epsilon)U_1(rsrs_{w-}, br_{w-}) > (w - \epsilon)U_1(rsrs_{w+}, p) + (1 - w + \epsilon)U_1(rsrs_{w+}, br_{w+})$$

(8)

(9)
We can manipulate Eq. 8 to get
\[
(w - \epsilon)U_1(rsrs_{w+}, p) + (1 - w + \epsilon)U_1(rsrs_{w+}, br_{w+}) \\
+ 2\epsilon((U_1(rsrs_{w+}, p) - U_1(rsrs_{w-}, p)) - (U_1(rsrs_{w+}, br_{w+}) \\
- U_1(rsrs_{w-}, br_{w-}))) > (w - \epsilon)U_1(rsrs_{w-}, p) \\
+ (1 - w + \epsilon)U_1(rsrs_{w-}, br_{w-})
\] (10)

For this and Eq. 9 to be true, we must have
\[
2\epsilon((U_1(rsrs_{w+}, p) - U_1(rsrs_{w-}, p)) > \\
(U_1(rsrs_{w+}, br_{w+}) - U_1(rsrs_{w-}, br_{w-}))
\] (11)

We know that $U_1(rsrs_{w+}, br_{w+}) - U_1(rsrs_{w-}, br_{w-})$ and that $U_1(rsrs_{w+}, p) - U_1(rsrs_{w-}, p)$ have different signs. If the first term is positive and the second negative, then Eq. 11 will be false, so it must be that $U_1(rsrs_{w+}, p) - U_1(rsrs_{w-}, p)$ is positive and $U_1(rsrs_{w+}, br_{w+}) - U_1(rsrs_{w-}, br_{w-})$ is negative. □

**Theorem.** The ratio of the gain against the prediction to the loss against a best-responding opponent is $\frac{1}{w} - \epsilon$.

\[
\frac{U_1(rsrs_{w+}, p) - U_1(rsrs_{w-}, p)}{U_1(rsrs_{w+}, br_{w+}) - U_1(rsrs_{w-}, br_{w-})} = \frac{1 - w}{w}
\]

**Proof.** From how $rsrs_{w+}$ is calculated we have
\[
U_1^{w+}(rsrs_{w+}, br_{w+}) > U_1^{w-}(rsrs_{w-}, br_{w-})
\]
where $U_1^{w+}(rsrs_{w+}, br_{w+})$ is the expected value of playing $rsrs_{w+}$ against a best-responding opponent in the modified game $U^{w+}$. Similarly
\[
U_1^{w-}(rsrs_{w-}, br_{w-}) > U_1^{w-}(rsrs_{w+}, br_{w+})
\]

From how $U^w$ is constructed we have
\[
\begin{align*}
(w + \epsilon)U_1(rsrs_{w+}, p) + (1 - w + \epsilon)U_1(rsrs_{w+}, br_{w+}) & > \\
(w + \epsilon)U_1(rsrs_{w-}, p) + (1 - w + \epsilon)U_1(rsrs_{w-}, br_{w-})
\end{align*}
\] (12)

\[
\begin{align*}
(w - \epsilon)U_1(rsrs_{w-}, p) + (1 - w + \epsilon)U_1(rsrs_{w-}, br_{w-}) & > \\
(w - \epsilon)U_1(rsrs_{w+}, p) + (1 - w + \epsilon)U_1(rsrs_{w+}, br_{w+})
\end{align*}
\] (13)

By rearranging terms we have:
\[
\frac{U_1(rsrs_{w+}, p) - U_1(rsrs_{w-}, p)}{U_1(rsrs_{w+}, br_{w-}) - U_1(rsrs_{w-}, br_{w-})} > \frac{1 - w - \epsilon}{w + \epsilon}
\] (14)

and
\[
\frac{U_1(rsrs_{w+}, p) - U_1(rsrs_{w-}, p)}{U_1(rsrs_{w+}, br_{w-}) - U_1(rsrs_{w-}, br_{w-})} < \frac{1 - w + \epsilon}{w - \epsilon}
\] (15)

Eqs. 14 and 15 provide the lower and upper bound. By taking the limit as $\epsilon$ converges to 0 we prove the theorem. □

**Learning Weight Values**
Calculating RSRS is straightforward given a weight assigned to the prediction, but it is not simple to see what that weight should be. RSRS assumes that the opponent will best respond if the prediction is incorrect, so we calculate a weight value by calculating the relative probabilities of the opponent playing according to the prediction and the opponent playing a best response.

We can easily calculate the probability of the sequence of play given that the prediction is correct. Calculating the probability of the sequence of play given that the opponent is best-responding is more complicated. The naive approach would assign a probability of 1 or 0 (either the opponent has played a best response in every previous game, or they have not). That approach is easy to deceive – for example, an opponent which consistently plays the second-best response to our strategy would not be considered to be best-responding.

We adopt a model in which the opponent plays according to an exponential response function to their expected payoff against our chosen strategy. Given an agent strategy $s_1 \in \Delta M_1$, the opponent’s exponentially weighted response is
\[
P(m_2^j) = \frac{e^{\lambda U_2(s_1, m_2^j)}}{\sum_{m_2 \in M_2} e^{\lambda U_2(s_1, m_2)}}
\]

where $\lambda$ describes how strongly the opponent chooses moves that perform better against the agent strategy. $\lambda = 0$ describes an opponent that plays uniformly at random. The higher the $\lambda$ value, the stronger the opponent’s preference for higher expected payoff moves (a negative $\lambda$ value indicates an opponent attempting to reduce their own payoff).

We calculate a value for $\lambda$ by finding the value with the maximum likelihood for the prior actions of the opponent. If all the observations have been best responses, the probability maximizing value will be infinity. To avoid this we introduce a smoothing observation in which the opponent selected an expected value of .9 out of options (0,.9,1).

We use $\lambda$ to compute the probability that a best-responding opponent played the observed move. This allows us to compute the relative probability of a best-responding opponent vs. an opponent playing according to the prediction. We use that value to determine the probability weight to use with RNR and RSRS. Steps are in Algorithm 1.

**Results in Rock/Spock/Paper/Lizard/Scissors**
We report here results we obtained in our experiments with Rock/Spock/Paper/Lizard/Scissors. Similar results with Battle of the Sexes, Chicken, Traveller’s Dilemma, Stag Hunt, and several other games are omitted for lack of space. In the experiments, agents play a sequence of 100 games. After each round they observe the move played by the opponent, but not the mixed strategy from which that move was drawn. Agents predict the opponent using fictitious play: they assume a stationary opponent playing according to a probability distribution drawn from a uniform Dirichlet distribution, perform Bayesian updates based on the observed moves, and use the expected value of the posterior as the prediction. Agents also compute the relative probability of the prediction and a best-responding opponent.

Figure 4 shows performance against three opponents, an omniscient best-responding opponent (a perfect Stackelberg follower), a worst-case opponent (one that plays the worst possible move for the agent), and a simple learning opponent (one that watches the agent play for 50 moves, and then plays a best response to that for the rest of the game). The agent makes a prediction using fictitious play and responds in one of five ways: (1) play a best response, (2) play a RNR
Best-responding opponent that trait. SPS treats the situation as worst case, and achieves responding, after which the agent plays to take advantage of 10–20 observations to determine that the opponent is best-responding, but RSR achieves a better outcome because it does not require its own strategy to be a best-response to the Stackelberg response perform equally well – it takes using the calculated weight, (3) play using SPS, (4) play a RSR using the calculated weight, and (5) play a RSRS using the calculated weight and SPS to determine risk factors.

Against a best-responding opponent, approaches based on the Stackelberg response perform equally well – it takes 10–20 observations to determine that the opponent is best-responding, after which the agent plays to take advantage of that trait. SPS treats the situation as worst case, and achieves the value of the game. RNR uses the same weight value as RSRS but achieves a worse outcome. Both agents are trying to find a strategy which performs well when the opponent is best-responding, but RSR achieves a better outcome because it does not require its own strategy to be a best-response to the opponent strategy.

Approaches which incorporate SPS quickly detect a worst-case opponent and get the value of the game (which is the best possible outcome in this situation). Of the agents that don’t use SPS, RNR does better than RSR (without safety) because assuming a best-responding opponent is inaccurate. The switching opponent is difficult for all agents because the predictor doesn’t handle the change in strategy at turn 50. Stackelberg-based strategies show a slight improvement in recovery after the switch, perhaps because the strategy it switches to initially resembles a best-response.

Conclusions and Future Work
We have presented Restricted Stackelberg Response with Safety (RSRS), a new method for choosing a strategy in a general-sum normal form game. We have shown experimentally that RSRS, by appropriate choice of parameter values, can take advantage of a prediction of opponent behavior while performing well against opponents that can predict its actions. Worst-case performance guarantees can be made by using SPS to choose values for the risk. Our results show that the weight-learning algorithm is effective, but it would be important to develop a more solid theoretical basis. We also plan to explore the performance of RSRS in self-play.
References


