A Case for Simulated Self-Play in Decision Models with Learning

Christoper Zosh*

April 19, 2025

Abstract

While there is an extensive history of bringing decision theories with learning to lab data, such models have been plagued with inadequate assumptions about the information players know before the first round of play. To solve this problem, I discuss the notion of Simulated Self-Play (SSP), in which agents play simulated rounds of the game against themselves to develop intuition about the nature of the game before the first round of play. Although some existing models of artificial intelligence have utilized self-play to achieve high performing solutions to some fairly complex problems (e.g. Alpha Zero playing Chess and Go), its exploration as a cognitive parameter when modeling human behavior has been relatively unexplored. First, I make the case that SSP improves theoretical coherence by discussing a number of common alternative assumptions (uniform / no priors, fitted priors, and burned-in priors), some of their a priori issues, and how Simulated Self-Play addresses many of them in a parsimonious way. Next, I evaluate the empirical value of SSP by implementing a simple learning model using priors formed via SSP and the alternatives and then compare their performance at predicting out-of-sample play in variations of the Beauty Contest game. I find that Simulated Self-Play performs as well or better than all of the aforementioned alternatives.

1 Introduction

As Behavioral Economics has risen to its current status as a legitimate and recognized subfield within Economics, there is perhaps more interest in our field now than

*Economics, Binghamton University

ever before in developing robust models of 'boundedly rational' decision making. What these models should look like is still a subject of much debate, with many of these discussions centered around how such models compare to rational decision making. There have been many arguments for the use of rational agents in models as reasonable proxies of long-run decision making. Lucas [1986] famously provides one such justification for models using rational expectations, arguing that they can be thought of as

> "...steady states of some adaptive process (where) decision rules (have been) found to work over a range of situations and hence are no longer revised appreciably as more experience accumulates."

Taking this interpretation of rational decision making at face value, it seems to indicate that for every rational strategy, there should be an underlying learning model from which the strategy emerges. Such boundedly rational models of decision making, if valid and fairly robust, could also prove vital in understanding when, how, and how frequently various social and economics systems arrive at different equilibria. While some papers have aimed to explore to what extent the long-run play by agents with learning algorithms match Nash play, the primary focus of modern work is on exploring to what extent play by 'agents' using such learning models match or forecast actual human behavior, including famously Roth and Erev [1995] and Camerer and Ho [1999].

Experience-Weighted Attraction (EWA) and some simple Reinforcement Learning (RL) models are very present in modern work, having been proposed by some as reasonable and fairly robust models of boundedly rational decision making with learning (Chen and Du [2017]). Case-Based Decision Theory (CBDT) has also been shown to perform fairly well in a variety of contexts (Gilboa and Schmeidler [1995], Pape and Kurtz [2013], and Guilfoos and Pape [2015]). For all such learning models, assumptions must be made about the information initially known before the first round of play occurs. Rather than exploring how such models compare directly, this paper focuses on both the implications of assumptions about initial information and how such assumptions affect model performance at forecasting out-of-sample play.

First, I discuss some ways in which initial beliefs are constructed for such models: assuming priors over actions are Uniform, treating priors as Free parameters, or Burning-in initial beliefs (as is done in some agent-based models). I also highlight some of the fairly undesirable implications of these assumptions. I then introduce the concept of Simulated Self-Play (SSP) as an alternative method for establishing initial beliefs, in which each learner plays hypothetical rounds of the game against themselves a number of times before the first round of actual play starts. Such

processes are known to be capable of achieving fairly high performing solutions to some very tough problems in the artificial intelligence world. One such example is Google's Alpha Zero, which learned to play Chess at an extremely high level primarily through repeated self-play. While SSP can solve problems well, it still remains unclear to what extent SSP is useful for modeling human behavior. There is some intuition for its value in such contexts, however. When introduced to the rules of a new game, it is not an uncommon experience to run through some example scenarios to form a basic understanding about the game and its dynamics. Further exploration of such examples may reveal strategies which perform well. I argue that SSP is a useful model of this human process which can be encoded for virtually any model of learning.

The remainder of this paper is organized as follows. In Section 2, I detail SSP and three alternative assumptions which guide how to initiate learning models, discussing some of the undesirable implications each comes with. In Section 3, I discuss some reasons why SSP may prove desirable over alternatives, as it addresses a number of their implied problems. In Section 4, I discuss all details of the empirical exercise conducted in this paper, testing a simple RL model's ability to forecast out-of-sample play in variations of the Beauty Contest Game (BCG) under each of these four initial information assumptions. In Section 5, I share the results of the empirical exercise, which finds burn-in and SSP to far outperform alternatives, with SSP performing slightly better. In the final two Sections, I discuss some open questions and potential future avenues for this work and conclude with the implications of what has been learned.

2 Initial Information Assumptions

Below I detail Simulated Self-Play (SSP) and a number of common alternatives can be used to form initial set of beliefs learning agents start with. Agents using SSP form their initial beliefs by playing a few imaginary rounds of the game against themselves with the aim of both understanding the mapping of actions to payoffs and exploring what strategies work well before the first round of play. I also detail three alternatives, which are by far the most common ways to initialize learning models: using uniform priors, fitting priors, and burning-in priors. I focus in particular on some of the problematic implications of these assumptions and how SSP aims to solve these problems in a parsimonious way, abstracting away from the mechanical details for now. Each will be revisited in Section 4.3, in which I show how these methods are encoded in the computational model.

2.1 No Information / Uniform Priors

Perhaps the most common way to initiate a learning model is to assume the decision-maker has no information before the first round of play. For the many models which use a form of attraction (Reinforcement Learning, Experience Weighted Attraction) or priors (e.g. Bayesian learning), this is encoded as all actions having the same initial attraction/weight. In other models which do not retain a set of attractions or priors, like in Case-Based Reasoning, this appears as an agent starting with an empty memory vector. So what are the problems with this assumption?

First, agents starting with no information have neither any beliefs about what their opponent will play nor do they start with any information about how actions map to payoffs. Round one of the game is a complete black box for these agents. They pick actions without any knowledge of how the game looks, they may observe what their opponents have picked, and then they wait to see what payoffs come out of the box. This is particularly a poor assumption when aiming to explain behavior of lab participants as a great deal of effort goes into making sure the players understand the game to at least some degree. A logical implication of this assumption is that we should expect lab participants to behave precisely the same whether or not they are shown the game matrix with the payoffs that correspond to each set of actions, since whether or not they are shown this matrix before the game starts, this information is not encoded into their decision making. Another implication if this assumption holds is that the behavior in the first round of any two games with the same number of actions and players is expected to be identical, regardless of what the payoffs are in the matrix. Yet another fairly unlikely implication is that participants playing a 2x2 game where one action is clearly inferior (i.e. strictly dominated) and potentially even very harmful to both players will be played with equal probability in expectation turn one.

Perhaps unsurprisingly, models utilizing this assumption often have a poor ability to explain early rounds of play. This remains true in the empirical exercise later in this paper. For situations where both action space is small and number of periods is very large (e.g. a 100 rounds of a 2x2 game), it may appear the issue uniform priors imposes is minimal as the influence of poorly fitted early periods on overall fitness shrinks when the number of rounds of play increases. While it is true that in such contexts, the effect of the no information / uniform priors assumption may be small, such findings do not say much about how well the learning model specified performs as a model of decision-making more generally, accounting for games where the number of rounds may be small and/or the action set is sufficiently large such that manual exploration would take a non-trivial amount of time.

2.2 Fitted Priors

Another common way to initiate a learning model is to assume that all decision makers enter with the same set of initial attractions/priors about actions before the first round of play. Rather than deriving these attractions from the game matrix itself, these initial attractions are typically treated each as a free parameter to fit in the model. In this way, these can be thought of as exogenously given attractions. So what are the problems with this assumption?

First off, such a method is not often feasible unless dealing with very small action sets because each action attraction is typically treated as its own free parameter. One can try to reduce the number of free parameters introduced by binning the attractions in nearby actions to have the same initial level (as was done in Chen and Du [2017]), but this only works games with actions that are well ordered. Second, such methods do not have a clearly tidy analogue in learning models which do not carry a set of attractions through periods like Case-Based Reasoning (though something similar was tried in Guilfoos and Pape [2015]). Finally, and perhaps most importantly, it is unclear to what degree such fitted parameters tell us something about behavior more generally. These parameters are not derived from features of the games themselves but rather are exogenously given. This means if we look at a slightly different version of the same game, where perhaps the payoffs in the matrix are slightly adjusted but all else remains the same, it is unclear to what extent, if at all, such parameters can help inform how agents will act in the new context. Taking this one step further, if aiming use this model to behavior in a similar but distinct game where the action sets between the games are not directly analogous, it is unclear how these fitted initial attractions can be used to inform us in this new context if at all. So, broadly, the extent to which fitting priors can tell us something about behavior which is generalizable or externally valid is unclear. As will be seen in the empirical exercise later, the case that such parameters tell us something useful in different versions of the game where the action sets are precisely the same is suspect.

2.3 Burned-In Priors

Burning-in a model is a term often used in agent-based modeling circles. To burn-in a model which would run for R rounds, you simply run it b+R rounds and then drop the first T rounds of play. When comparing model output to data, what is actually being compared is model output from rounds b+1, ... b+R and data from rounds 1,...,R. In general, this allows the modeler to initiate their model starting in a sort of mid-run or long-run state. In the context of models with learning agents specifically, this can be thought of as agents entering with some experience playing this game.

Interestingly, this allows agents to enter with different initial attractions/priors, since they will accrue different experiences during these burn-in 'practice' rounds with each other. When bringing such a model to data, the number of rounds used to burn-in the model can be treated as a free parameter which is denoted b. In many ways, burn-in solves the problems of the uniform priors / no information assumption in a much more parsimonious way than fitting attractions directly. Unlike the fitted individual attractions, the fitted burn-in parameter also retains interpretability across contexts, as it essentially is a parameter which corresponds the amount of experiences the players have playing the game with each other. When changes are made to the action set or the payoffs in the game, such changes can easily be accounted for in new predictions. So what then is the problems with this assumption?

The primary problem arises from the fact that a burn-in requires agents to play each other for b rounds. First, this means that all of their attractions, while they may differ, must correspond to the same set of events. For example, if players play a two player prisoner's dilemma and form initial beliefs using a burn-in, it will not be the case that one player believes repeated Coorperation is likely while the other believes that all players will probably just play Defect, as these beliefs correspond to different events. So the first implication of a burn-in is that while agents can enter with different priors, the set of possible combinations of priors agents can enter with are constrained only to the set of priors which correspond to shared experiences. Taking a step back, this doesn't make much sense when, particularly in a lab setting, these players have not yet played this game with each other. A slightly different though very related problem arises from the fact that, since these agents are playing rounds against each other before the first round of play, agents can quite literally see what their opponents have and how their opponent plays the game. This is a fairly strong and unrealistic assumption about what players, particularly in a lab setting, should know about their opponents before play starts. What is desired is a sort of burn-in where agents do not directly learn about their opponents before the first round of play and where the attractions/priors they enter with are not constrained to the set of events which must be shared between all agents. As we is shown in the next couple sections, Simulated Self-Play provides just that.

2.4 Priors from Simulated Self-Play

Simulated Self-Play (SSP) is very similar to a burn-in in many ways. Just like a burn-in, b rounds of 'pre-period' play are executed. However, unlike a burn-in, each agent using SSP will play these b rounds against themselves as if they are all the players in the game simultaneously. This can be thought of as each

agent, after seeing the rules of the game, working through a number of example cases of the game in their head before choosing an action on the first turn. As b increases, agents gain a more sophisticated understanding both of the mapping of actions to payoffs and of what strategies perform well by continually trying to improve on their previous strategies by playing more rounds against themselves. Importantly, these repeated games against oneself do not directly incorporate any information from actual play against their opponents. Intuitively, this means agents play strategies that worked well against themselves in their mental self-play games in the first round of play and then adjust their strategies over rounds of play with other players based on the experiences they have with their opponents. This also means agents can enter with beliefs consistent with different events. Running with the prisoner's dilemma example, this means that one agent thinking about the game can come to the conclusion that repeated cooperation is a real possibility and enters the game optimistically while another player could enter the game having come to the conclusion that always playing D seems to work pretty well.

3 An A Priori Case for Simulated Self-Play

Above I discussed a number of common assumptions about the initial information decision theories with learning are instantiated with. I also highlighted a number of issues presented with each, and concluded with the proposed alternative of Simulated Self-Play. So how does Simulated Self-Play address the issues present when assuming the alternatives?

As discussed in Section 2.1, assuming no information is known initially has a number of problematic implications, perhaps the most troubling of which is complete indifference to actions in every game, even when actions are strictly dominated and harmful to everyone. I also mentioned in Section 2.2 that what directly fitted action attractions can tell us about similar but distinct games is unclear, and sometimes such fitted parameters are simply incompatible (for example, when a small, trivial action is added to the action space). Unlike in the cases of Uniform or Fitted priors, SSP derives attractions using repeated play of the game itself. This means that the predictions of initial attractions using parameter b account for differences in the games presented to the agents and retains interpretability across games which can vary in action set size, payoffs, player count, etc. Further, SSP solves this problem in a much more parsimonious way than fitting priors directly. While moving from uniform priors to SSP introduces just one parameter b, fitted priors requires adding a number of parameters which grows proportionally with the size of the action set.

In Section 2.3, I discuss how burning-in priors also solves many of the problems

that arise when using Uniform or Fitted priors. However, it introduces a new problem of giving players experiences with each other that we know, based on how the experiment is set up in the laboratory, they should not have. Even if we accept this fact, taking burn-in as an imperfect but sufficient proxy, we know that the set of priors which agents can enter with is heavily biased, since agents cannot enter with priors which are not compatible to the same set of mutual events. We also rule out any dynamics that might emerge as a result of having the 'wrong idea' about your opponent in early periods of play. By allowing agents to use SSP over burning-in priors, agents can form priors of various sophistication levels (governed by b) without playing each other directly and without being constrained to a subset of priors agents could actually (and reasonably may) enter the game with. And again, this comes without any additional parameters as both burn-in and SSP utilize the same parameter b to govern rounds of pre-play.

4 An Empirical Test of Simulated Self-Play

So far I have discussed why simulated self-play may be preferred a priori for its desirable properties and theoretical coherence. However, its validity as an assumption about human behavior remains to be seen. To address this, I propose an empirical exercise in which versions of a computational model where agents leverage a very simple reinforcement learning algorithm compete to forecast out-of-sample lab play, with each version of the model distinguished only by the assumed initial information agents are given. This exercise is performed using lab data collected on groups of various sizes playing versions of the Beauty Contest Game (BCG). For the remainder of the section, I break down each of the components of the exercise (the game, the data, the learning model, and the evaluation criteria) in turn.

4.1 The Beauty Contest Game

The first discussion of the Beauty Contest Game is often attributed to [Keynes, 1936], in which he describes a hypothetical contest in which, from a large number of photographs, contestants are asked to choose the most attractive candidates. All players who selected the most popular photos would then be eligible to receive a prize. This served as an analogy to stock market behavior, illustrating how the desirability of a stock could be highly influenced by ones beliefs about how much others value it. This game was later popularized by various experiments which aimed to explore the role of beliefs about others in boundedly rational behavior Nagel [1995], Duffy and Nagel [2012], Grosskopf and Nagel [2008].

For better or worse, modern implementations of the game are not quite as Keynes imagined then (though programming on NPR came close, asking listeners to select from a small number of cute animal videos). Instead, each player submits a number a_i from the interval [0, 100], which defines the action set Θ. Once the submitted choices A = $\{a_1, ..., a_n\}$ are collected, the target number is computed in the following way:

$$\tau = \rho * Agg(A) \tag{1}$$

where $\rho \in$ (0,1) (0.5 in our case) and $Agg(.)$ is some function which aggregates across the numbers chosen. This is commonly the mean choice, but taking the median and max of the choices has also been tried a number of times, including in the data I'll be using from Duffy and Nagel [2012].

The winner(s) are the player(s) who chose the number, $a_{i,t}$, which is (are) closest to the target τ, with tied players splitting the prize equally. Thus, players can maximize their payoff by choosing a number a_i closest to the target number, which again, is some fraction of the aggregate.

This game has a number of curious properties which make it fairly desirable for our context. First, the important role that learning plays in the context of this problem is evident. The stage game has a unique weakly dominant Nash equilibrium (NE) which, for all of the aggregation functions $Agg(.)$ and ρ mentioned above, is for all players to choose the minimum choice 0 when there are greater than two players playing. Intuitively, there cannot be a symmetric NE greater than the minimum since any player splitting the prize with others would benefit from reducing their choice slightly, undercutting the choices of the group and gaining the remainder of the prize for themselves. Despite having a unique weakly dominant NE, in practice players often choose higher choices initially and approach the NE after a number of rounds. The speed of this approach to NE can not only on features of the game, but also on very early rounds of play. This has made it the subject of a number of papers investigating learning, boundedly rational response, and the apparent path dependence of play in such games Nagel [1995], Duffy and Nagel [2012], Grosskopf and Nagel [2008]. Second, this is one of the few canonical games in economics with an action set which is non-trivially large. Typically, players are asked to choose a number from the interval [0,100], which means (if the action set is discretized at the level of integers) there are 101 options for players to choose from each round. With an action set of this size, initial beliefs about the performance of these actions plays an important role, as manual exploration of the entire action set is not feasible in ten or less rounds of play. This makes this problem ideal for testing assumptions about initial beliefs / information.

4.2 Data

As mentioned above, I utilize lab data introduced in Duffy and Nagel [2012] which contains the per-round choices of participants playing variations of the repeated BCG. The dataset contains 868 data points in total, with trials varying in number of participants (N), number of rounds (Rounds), and how the choices are aggregated (Agg(.)) to produce the target number τ_t each round of play. These trials are summarized in the table below:

Session	p	N	Agg(.)	Rounds
1	0.5	15	Median	4
2	0.5	15	Median	4
3	0.5	13	Median	4
4	0.5	13	Median	10
5	0.5	16	Mean	4
6	0.5	14	Mean	4
7	0.5	15	Mean	4
8	0.5	14	Mean	10
9	0.5	15	Max	4
10	0.5	15	Max	4
11	0.5	15	Max	4
12	0.5	15	Max	10

Table 1: Lab Data from Duffy and Nagel [2012]

These variations in the game are an important feature of this data, as it allow us to test the degree to which our initial information assumptions tell us something about behavior more generally. As we will see later in the evaluation criteria, we will break this dataset into two parts. Approximately $\frac{2}{3}$ of the data is used to train our learning models and the last $\frac{1}{3}$ is used to evaluate how well our models perform at forecasting human behavior in very similar but importantly distinct versions of the game.

4.3 Agent Learning

4.3.1 The Simple Reinforcement Learning Model

For the purposes of this paper, which focuses on the assumptions made about initial information in particular, I elected to use a learning algorithm which is simple, uses

very few parameters, and already has some proven ability to explain behavior in lab settings. Roth and Erev [1995] propose a simple 1-parameter reinforcement learning model which they demonstrate outperforms NE in predicting play in 12 different 2x2 games. This work is extended in Erev and Roth [1998], adding two such behavioral parameters. The agents in my model use a simplified version of this reinforcement learning algorithm which, excluding the parameters required to establish the initial priors of the agents, uses only one parameter. I detail this simple learning algorithm below.

Initialization:

Each agent starts with a vector Γ of 'attractions' to each action. Since in this setting there are 101 actions to choose from (integers from [0,100]), agents must start with a set of 101 attractions.

$$\Gamma_{i,t=-1} = \{\gamma_{0,i,t=-1}, ..., \gamma_{100,i,t=-1}\} \tag{2}$$

The initial value of these attractions (which can be thought of similarly as priors in a Bayesian framework) are determined by our assumptions about initial information. In Erev and Roth [1998], this value is determined by a parameter S, which represents the strength of agents' priors. The particular encoding we will use to determine Γ depends on which assumptions about initial information we are using. Each encoding of these assumptions will be detailed in turn in the subsections that follow. For now, take the initial $\Gamma_{i,t=-1}$ as given.

Round Behavior:

Each round, each agent in a simulated trial will choose an action $a_{i,t} \in \{0, ..., 100\}$ randomly, weighted by the action's attraction level. The likelihood agent i will choose a particular action $a_{i,t} = x$ at time t is directly proportional to its size in the attraction set. Formally:

$$Prob(a_{i,t} = x) = \begin{cases} \frac{\gamma_{x,i,t}}{\sum_{j=0}^{100} \gamma_{j,i,t}} & \text{if } x \in \{0, ..., 100\} \\ 0 & \text{otherwise} \end{cases} \tag{3}$$

Once actions are chosen, the actions are fed into the simulated game, which takes agent actions each period $\{a_{1,t}, ..., a_{N,t}\}$ are arguments, computes the target level τ, then returns payoffs to the agents $\{\pi_{1,t}, ..., \pi_{N,t}\}$. At the end of the round, each agent uses the experience of playing the round $\{a_{1,t}, \pi_{1,t}\}$ to update their attractions to each action for next round in the following way:

$$\gamma_{i,t+1}(\hat{a}) = \begin{cases} (1-R) * \gamma_{x,i,t} + \pi_{i,t} & \text{if } x = a_{i,t} \\ (1-R) * \gamma_{x,i,t} & \text{otherwise} \end{cases} \tag{4}$$

In words, the influence of all past attractions are reduced using the second free parameter R, Recency Bias. Then, the attraction associated with the chosen action this period $\gamma_{a_{i,t}}$ is adjusted additively by the payoff received this period $\pi_{i,t}$.

4.3.2 Encoding Uniform / No Priors

The simplest and often most common assumption in learning models is to assume no information. In many models, a logical jump is made that, in the face of this ambiguity, agents should be equally willing to select any action. To implement this in the learning model above, we simply set each initial attraction equal to some value S. Formally:

$$\Gamma_{i,t=-1} = \{\gamma_{0,i,t=-1} = S, ..., \gamma_{100,i,t=-1} = S\} \tag{5}$$

This is how initial priors are set up in Erev and Roth [1998], with the interpretation of S being the strength of initial priors. If S is very large, it will typically take more experiences to move away from uniform priors. Importantly, as seen above, all action attractions are assigned the same value S to start. Also note, this assumption adds just 1 parameter to the model.

4.3.3 Encoding Fitted Priors

Another alternative discussed above is to treat the initial attractions agents have to particular actions as free parameters to fit. Importantly, the initial set of attractions for all agents are identical, just like in the uniform priors case.

Recall in the base learning model, there are 101 attractions agents start with. One thought might be to fit all 101 as separate parameters, but this seems to be almost the definition of overfitting and would quite certainly yield poor results out of sample. Fitting this many parameters well could also prove immensely computationally expensive as the parameter space is massive, decreasing the odds of finding a global optimum in any reasonable amount of time. In the spirit of trying to give this assumption the best chance to succeed, I instead bin action attractions and fit those bins, as was done in Chen and Du [2017]. Initial attractions are broken into five bins of approximately 20 actions each (the first bin contains the one extra initial

attraction), each of which get a free parameter $S_1, ..., S_5$, which correspond to the starting values of the attractions in their bin. Formally:

$$\begin{aligned}\Gamma_{i,t=-1} = \{&\gamma_{0,i,t=-1} = S_1, ..., \gamma_{20,i,t=-1} = S_1 \\ &\gamma_{21,i,t=-1} = S_2, ..., \gamma_{40,i,t=-1} = S_2, \\ &\gamma_{41,i,t=-1} = S_3, ..., \gamma_{60,i,t=-1} = S_3, \\ &\gamma_{61,i,t=-1} = S_4, ..., \gamma_{80,i,t=-1} = S_4, \\ &\gamma_{81,i,t=-1} = S_5, ..., \gamma_{100,i,t=-1} = S_5\}\end{aligned} \tag{6}$$

This assumption adds 5 parameters to our model $S_1, .., S_5$, one for each bin.

4.3.4 Encoding Burned-In Priors

To implement assumed burned-in priors in our context, we simply initiate the model the same exact way as we do when there are no priors. However, we add one additional step; we have the agents play the repeated game b times, and take the attractions after those b runs to use as our initial attractions for agents in the actual runs we want to compare to data. Again, this is meant to simulate the fact that these agents have prior experiences or prior knowledge about how the game might work. If we denote the process of running the model r periods as $ABM(\theta, r)$, then the initial attractions for burned-in agents can be given as:

$$ABM(\theta, r = b * Rounds) \rightarrow \Gamma_{i,t=-1} \tag{7}$$

with

$$\hat{\Gamma}_{i,t=-1} = \{\gamma_{0,i,t=-1} = S, ..., \gamma_{100,i,t=-1} = S\} \tag{8}$$

This allows for agents to start with different initial attractions (though these the space of initial attractions possible for agents is constrained to those which are derived from the same mutual experiences). Also note that this adds two parameters to the model: S and b. S, as before, represents the strength of initial priors. b on the other hand represents the number of burn-in rounds the agents go through before play begins.

4.3.5 Encoding Priors using Simulated Self-Play

Very similar to burned-in priors, we once again derive our initial action attractions from rounds of past play, which themselves are initiated with uniform priors. The one distinction, however, is that these runs are computed for each agent separately, playing a special version of the game where instead of playing the other agents, they

play the game against themselves as if they are all players at once, and they accrue all such experiences. Importantly, as noted before, this allows agents to enter with different priors that, unlike burned-in priors, do not need to conform to the same mutual experiences. If we denote this special version of the simulation where agents play themselves $A\tilde{B}M_i(\theta, r)$, then the initial attractions for agents using simulated self play is given by:

$$A\tilde{B}M_i(\theta, r = b * Rounds) \rightarrow \Gamma_{i,t=-1} \tag{9}$$

where once again

$$\hat{\Gamma}_{i,t=-1} = \{\gamma_{0,i,t=-1} = S, ..., \gamma_{100,i,t=-1} = S\} \tag{8}$$

Again, this model adds two parameters S and b, where S is the strength of priors and b is the number of rounds of self-play agents engage in to form their initial beliefs.

4.4 Evaluation Criteria

First, I divide up the dataset (as mentioned earlier in Section 4.2) into training and evaluation criteria in the following way, with white cells in the training dataset and gray in the evaluation dataset.

This loop of behavior is executed once for each agent in each round of play within a given run, mirroring the precise problems faced by agents in the lab experiment. For example, in session 5, sixteen participants play four rounds of the BCG using a mean aggregation function. In my simulation, sixteen simulated agents play four rounds of the BCG with a mean aggregation function, making decisions each round as detailed above, initialized with one of the four initial information assumptions. This digital mirroring ensures the circumstances simulated agents face are meaningfully comparable to the circumstances that were faced by the participants in the lab.

In principle, the evaluation set serves as something to compare our model predictions to which has not been trained on, but which are in many ways similar (though distinct) from the sessions contained in the training data. This is important as it helps guard against over-fitting issues. The intuition goes, if a model specification is very flexible (e.g. it has 10,000 parameters), it likely can achieve very low levels of loss when compared to the training data. However, that over-fitted model likely will not generalize well when trying to explain data which it has not been able to train on.

Each of these four models have different sets of parameters which are listed in the table below:

Session	p	N	Agg(.)	Rounds
1	0.5	15	Median	4
2	0.5	15	Median	4
3	0.5	13	Median	4
4	0.5	13	Median	10
5	0.5	16	Mean	4
6	0.5	14	Mean	4
7	0.5	15	Mean	4
8	0.5	14	Mean	10
9	0.5	15	Max	4
10	0.5	15	Max	4
11	0.5	15	Max	4
12	0.5	15	Max	10

Table 2: Training and Evaluation Data

Model	**Parameters**
Uniform Attractions	R, S
Fitted Attractions	R, S_1, S_2, S_3, S_4, S_5
Burned-In Attractions	R, S, b
SSP Attractions	R, S, b

Table 3: Best-fitted parameters for each model.

A loss function must be specified both to find best-fitting parameters for each version of the learning model when applied to training data and to evaluate how well each model with best-fitted parameters emulates play in the evaluation set. Following the paradigm of Simulated Method of Moments, I use a fitness function which aims to capture the weighted difference in the first two moments of choices between agents in the simulation and players in the lab. This is given as follows:

$$Loss(ABM(\theta, r)) = \sum_{s \in D} \sum_{t=0}^{Rounds} \Big[(\overline{y_{s,t}} - \overline{\hat{y_{s,t}}}(\theta, r))^2 + \alpha(Var_t(y_{s,t}) - Var_t(\hat{y_{s,t}}(\theta, r)))^2 \Big] \tag{10}$$

where s is the session number, D is either training or evaluation data, $y_{i,s,t}$ is an

observed choice in D, $\hat{y_{i,s,t}}(\theta, r)$ a choice from model output, and α is the relative weight given to the second moment in the loss function. I set $\alpha = 0.05$. Note since $\overline{\hat{y_{s,t}}}(\theta, r)$ and $Var_t(\hat{y_{s,t}}(\theta, r))$ are from a computational model which is stochastic and could be fairly path dependent, these values must be computed over a number of model runs. Both moments of model output take an argument r which corresponds to the number of model runs used to compute the expected outcomes of these moments.

As is often the case with computational models, the best-fitting parameters for each model (i.e. the parameters which minimize loss when compared to the training set) requires the use of some algorithmic searching of the parameter space. I utilize a method known as behavioral search. While such an algorithm does not guarantee the globally optimal solution, there are no alternatives which should perform better a priori and all models are equally subjected to this form of optimization. Taking these four fitted models, a comparison of their ability to forecast the out-of-sample behavior in our evaluation data can now be made.

5 Results

The parameters which best fit the training data associated with each model can be found in the table below.

Model	R	S	S_1	S_2	S_3	S_4	S_5	b
Uniform Attractions	0.99	0.01	-	-	-	-	-	-
Fitted Attractions	0.822	-	0.01	55.468	51.603	10.794	5.254	-
Burned-In Attractions	0.658	38.040	-	-	-	-	-	4
SSP Attractions	0.732	68.068	-	-	-	-	-	4

Table 4: Fitted model parameters on training data

Upon quick inspection, there are a few things worth noting. First, we may notice that the models using burned-in and SSP attractions have fairly similar fitted parameter values, with both utilizing 4 rounds of pre-play and fairly close levels of recency bias. The higher strength of priors S for SSP may also explained in part by the fact that SSP players play the game from all perspectives during the pre-play rounds, incorporating how well actions perform from all perspectives of the round. In contrast, burn-in players only incorporate experiences from their own actions. Since SSP players incorporate more information per round of pre-play, a higher S may be required to retain a similar level of willingness to explore new strategies post pre-play rounds.

Next, we may notice that the fitted attractions model seems to put much more initial attraction into S_2 and S_3, which correspond to actions $\{21, ..., 60\}$ while placing very little weight at the extremes. This seems fairly reasonable and is in-line with what is often seen in such games.

Finally, it seems as though in the uniform priors case, S is nearly minimized and R is nearly maximized. A high R corresponds to rapid reduction in the influence of old experiences on current attraction to actions. A small S indicates that the belief that all actions perform equally well is very weak. It seems the best the uniform priors version of the model can do to replicate the behavior of agents in the training set is to choose effectively uniform randomly until some payoff is earned, after which it will continue to pick that action with near certainty. Once that action ceases to return a positive payoff, it will rapidly return to drawing actions in a nearly uniform random manner once again until one happens to return a positive payoff again. While there is some sense to this solution, it seems quite erratic. As we will see later, this model performs fairly poorly at forecasting play in the evaluation set.

Next, we see the results of the forecasts of each model, reporting both the loss and the relative size of loss when compared to the model using SSP. This is summarized both in the table and plot below.

Model	**Loss**	**Loss / SPP Loss**
Uniform Attractions	42.757	2.392
Fitted Attractions	33.347	1.866
Burned-In Attractions	18.814	1.05
SSP Attractions	17.871	1

Table 5: Fitted model loss on evaluation data

First, it can be seen that SSP and Burn-in perform similarly well and outperform the other two alternatives, with Self-play performing slightly better in this instance, though it cannot be said if this difference in performance is statistically significant. Additionally, the learning model utilizing Uniform priors performs the worst by a reasonable margin, with more than twice the amount of loss in its forecast over SSP and burn-in. Given the BCG has 101 actions to choose from, entering the game with no prior information (including no sense of what the payoffs of the game even are) to guide decision making means the only way such agents can learn is through manual exploration of the massive action set. Given real-world participants play this game over either 4 or 10 rounds in a session, it seems unlikely that manual exploration alone can fully explain player behavior, and that's precisely what the poor performance of this model is telling us. Fitting priors directly also appears to

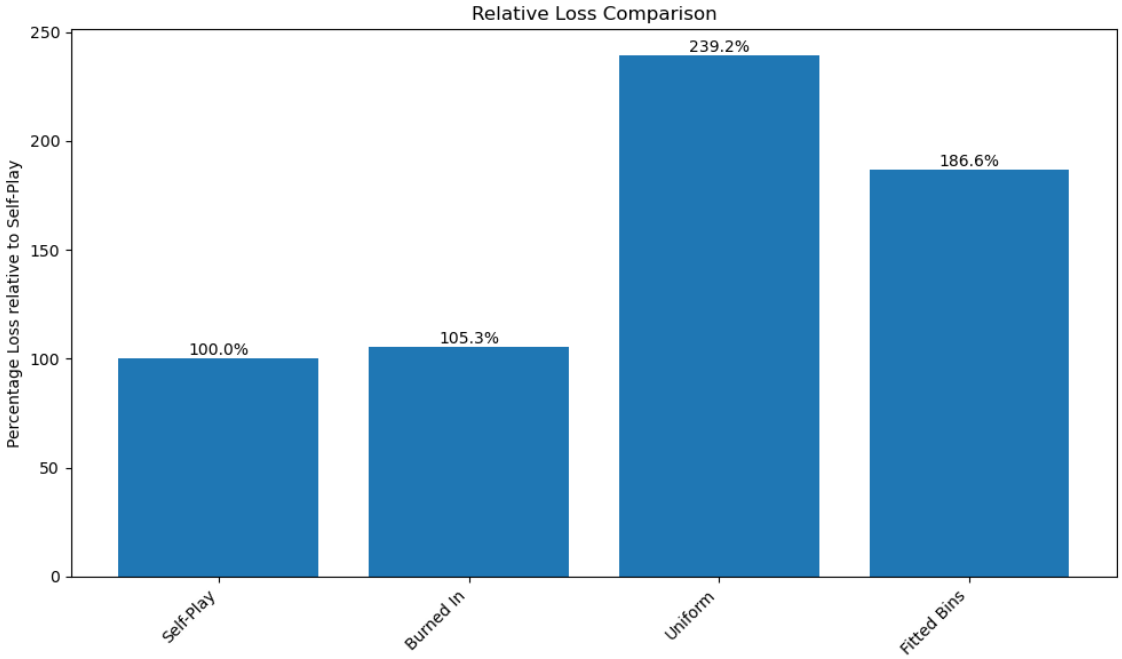

Figure 1: Best-fitted model loss in forecast of evaluation set play relative to SSP performance

perform worse than both burn-in and SSP, though better than Uniform priors. This poor performance may come as a surprise, but it may be the case that in this context where the evaluation set contains slightly different versions of the game, these initial attraction bins fitted on the training data are not as externally valid as one might hope for.

6 Discussion

This project was initially motivated by my own experience trying to understand why various decision theories with learning I had encountered seemed to under perform when trying to explain play in games with large action sets. Within the economics literature, the number of papers that bring reinforcement learning models to lab data with the goal of trying to understand decision making remains fairly small, though interest has been growing. Very few of those papers, however, apply such models to non-2x2 games. To test the validity of such models, exploration of their performance in less represented contexts needed, including games with non-trivially large action

sets. This paper, which explores how well variations of a simple reinforcement learning model can forecast out-of-sample play in the BCG as is just one small step in that direction.

These results also raise some additional questions about how to think about the role our assumptions play in constraining our models. One of the primary a priori arguments I make for SSP over burn-in is based on the fact that burn-in constrains the set of possible initial attractions agents can enter the game with, which in turn may be creating a sort of complex bias in outcomes we might expect from the model. Uniform attractions and Fitted attractions are likely even more constraining, as making these assumptions requires all agents to have precisely the same initial attractions. In many ways, the empirical exercise presented above can be thought of as a comparison of how well the same simple learning model performs when the initial conditions of that model are constrained in different ways, governed by various parameters. While a small difference in performance can be seen between the learning model using burn-in and SSP, it is not yet clear that the application of these assumptions about initial information to learning models used in another game should produce similar differences in performance. Intuitively, given SSP allows all n agents to enter with attractions which are consistent with their own set of events while burn-in requires all n agents to have attractions consistent with only one event, I expect the difference in possible initial conditions of the model to grow with player count or with the size of the action set, though there is no guarantee that these additional events lost produce outcomes which are qualitatively different. Further exploration of the effective difference these assumptions make in new contexts is needed.

Finally, it is important to note that the advantages of SSP do come at a computational cost. While burn-in requires running the model once for b rounds, the SSP requires running the model n times for b rounds, where n is agent count. SSP as described above could become less feasible computationally if dealing with models that have, for example, thousands of agents. However, an interesting solution potentially worth exploring in the future could be to generate initial attractions using SSP for, say 50 agents, and then initiate each of these thousands of agent with initial attractions drawn randomly with replacement from this set of 50. While this would constrain the set of possible combinations of initial beliefs agents can start with, it does less so than doing a burning-in the model.

7 Conclusion

Above I have presented a case for SSP as a reasonable way to approximate the information players start games with. I provide a number of reasons why we might prefer to use SSP a priori, as it solves a number of issues created when using alternative assumptions in a fairly parsimonious way. I also provide empirical evidence that for a simple learning model, using SSP to construct in initial beliefs agents enter the game with improves the model's ability to emulate out-of-sample play. I hope that this contribution inspires other analysts to consider the role that SSP could play as in their models of behavior and to demonstrate the importance of pushing the boundaries of modeling assumptions more generally.

References

Colin Camerer and Teck-Hua Ho. Experienced-weighted attraction learning in normal form games. Econometrica, 67(4):827–874, 1999. ISSN 00129682, 14680262.

Shu-Heng Chen and Ye-Rong Du. Heterogeneity in generalized reinforcement learning and its relation to cognitive ability. Cognitive Systems Research, 42:1–22, 2017. ISSN 1389-0417. doi: https://doi.org/10.1016/j.cogsys.2016.11.001. URL https://www.sciencedirect.com/science/article/pii/S1389041716300559.

John Duffy and Rosemarie Nagel. On the Robustness of Behaviour in Experimental 'Beauty Contest' Games*. The Economic Journal, 107(445):1684–1700, 01 2012. ISSN 0013-0133. doi: 10.1111/j.1468-0297.1997.tb00075.x.

Ido Erev and Alvin E Roth. Predicting how people play games: Reinforcement learning in experimental games with unique, mixed strategy equilibria. American Economic Review, 88(4):848–881, September 1998.

Itzhak Gilboa and David Schmeidler. Case-based decision theory. The Quarterly Journal of Economics, 110(3):605–639, 1995. ISSN 00335533, 15314650.

Brit Grosskopf and Rosemarie Nagel. The two-person beauty contest. Games and Economic Behavior, 62(1):93–99, 2008. ISSN 0899-8256. doi: https://doi.org/10.1016/j.geb.2007.03.004.

Todd Guilfoos and Andreas Pape. Predicting human cooperation in the prisoner's dilemma using case-based decision theory. Theory and Decision, 80, 04 2015. doi: 10.1007/s11238-015-9495-y.

John M. Keynes. The General Theory of Employment, Interest and Money. Macmillan, 1936. 14th edition, 1973.

Robert E. Lucas. Adaptive behavior and economic theory. The Journal of Business, 59(4):S401–S426, 1986. ISSN 00219398, 15375374.

Rosemarie Nagel. Unraveling in guessing games: An experimental study. The American Economic Review, 85(5):1313–1326, 1995. ISSN 00028282.

Andreas Duus Pape and Kenneth J Kurtz. Evaluating case-based decision theory: Predicting empirical patterns of human classification learning. Games and Economic Behavior, 82:52–65, 2013.

Alvin Roth and Ido Erev. Learning in extensive-form games: Experimental data and simple dynamic models in the intermediate term. Games and Economic Behavior, 8(1):164–212, 1995.