Bounded Rationality via Recursion

Maciej Łatek and Robert Axtell
Center for Social Complexity
Department of Computational Social Sciences
George Mason University
4400 University Dr., Fairfax, Virginia, U.S.A.
mlatek,rax222@gmu.edu

Bogumil Kaminski
Decision Support and Analysis Division
Institute of Econometrics
Warsaw School of Economics
Al. Niepodleglosci 162, Warsaw, Poland
bkamins@sgh.waw.pl

ABSTRACT

Current trends in model construction in the field of agent-based computational economics base behavior of agents on either game theoretic procedures (e.g. belief learning, fictitious play, Bayesian learning) or are inspired by artificial intelligence (e.g. reinforcement learning). Evidence from experiments with human subjects puts the first approach in doubt, whereas the second one imposes significant computational and memory requirements on agents.

In this paper, we introduce an efficient computational implementation of n-th order rationality using recursive simulation. An agent is n-th order rational if it determines its best response assuming that other agents are $(n-1)$-th order rational and zero-order agents behave according to a specified, non-strategic, rule. In recursive simulations, the simulated decision makers use simulation to inform their own decision making (search for best responses).

Our goal is to provide agent modelers with an off-the-shelf implementation of n-th order rationality that leads to model-consistent behaviors of agents, without requiring a learning phase. We extend two classic games (Shapley's fictitious play and Colonel Blotto) to illustrate aspects of the n-th order rationality concept as implemented in our framework.

Categories and Subject Descriptors

J.2 [**Social and Behavioral Sciences**]: Economics—*Game Theory*; I.2.11 [**Artificial Intelligence**]: Distributed Artificial Intelligence—*Multi-agent Systems*

General Terms

Algorithms,Economics,Cognition

Keywords

Recursive Agent-based Models, Multiagent Learning and Decision-making, Cognitive Architecture

1. INTRODUCTION

In John Maynard Keynes' most well-known book [17, Chapter 12] he describes the behavior of investors as follows:

> ... professional investment may be likened to those newspaper competitions in which the competitors have to pick out the six prettiest faces from a hundred photographs, the prize being awarded to the competitor whose choice most nearly corresponds to the average preferences of the competitors as a whole; so that each competitor has to pick, not those faces which he himself finds prettiest, but those which he thinks likeliest to catch the fancy of the other competitors, all of whom are looking at the problem from the same point of view. It is not a case of choosing those which, to the best of one's judgment, are really the prettiest, nor even those which average opinion genuinely thinks the prettiest. We have reached the third degree where we devote our intelligences to anticipating what average opinion expects the average opinion to be. And there are some, I believe, who practice the fourth, fifth and higher degrees.

This kind of rationality is called in economics n-th order rationality. A formal definition is presented in [21]. An agent is first-order rational if it calculates the best response to his beliefs about strategies of zero-order agents[1] and the state of the world. An agent is n-th order rational if it determines its best response assuming that the other agents are $(n-1)$-th order rational[2].

H. Simon [25] differentiates between agents capable of achieving rational outcomes deus ex machina (*substantive rationality*, which does not provide a plausible mechanism by which rational results might be achieved) and those operating according to feasible but potentially less capable heuristics (*procedural rationality*). Analytical game theory, with it's highly abstract concepts like perfect Bayesian equilibrium, is substantively rational for well-conditioned environments. However, wide classes of substantively rational solution concepts are computationally intractable and therefore are unlikely to be useful in reality[3]. A concept of n-th order rationality belongs to procedural rationality class, but aims to bridge procedural and substantive approaches.

Cite as: Bounded Rationality via Recursion, Maciej Łatek, Robert Axtell, Bogumil Kaminski, *Proc. of 8th Int. Conf. on Autonomous Agents and Multiagent Systems (AAMAS 2009),* Decker, Sichman, Sierra and Castelfranchi (eds.), May, 10–15, 2009, Budapest, Hungary, pp. 457–464

[1] As discussed later, the exact choice of behavioral rule for zero-order agent's is usually case-specific and could include random behavior, continuation of historical behavior (our default) or any non-strategic learning rule.

[2] The philosophy behind an iterated process of strategic thinking has been outlined in [2].

[3] See [1] for discussion of computational complexity of different concepts of market equilibrium in economics.

First, it permits models with different levels of player rationality (through the choice of n). Second, as shown in this paper, it can be efficiently implemented and solved even for complex environments.

As outlined in the next section, the concept of n-th order rationality has recently applied in economic models of simple environments with small n. However, in problems with complex environments and heterogeneous agents, tracking n-th order behaviors becomes analytically tedious and computational methods need to be applied.

In this paper, we introduce an efficient computational implementation of n-th order rationality using recursive simulation. Doing this, we demonstrate:

1. A structural design that enables introduction of n-th order rational agents into any agent-based model;

2. That agents using n-th order rationality behave in a model-consistent manner without requiring any learning phase[4];

3. Sensitivity analysis of model results with respect to the computational and cognitive complexity of agents is enabled by the n-th order rationality approach.

Lastly, recursive models based on n-th level rationality have a number of degrees of freedom (for example: rationality levels for all strategic agents), that can be calibrated using data. If this is accomplished, models perform *descriptive* role. By increasing the rationality level of an selected agent by 1, we generate guidelines with respect to how this agent *should* behave, therefore having n-th level rationality fill the *normative* role of a decision framework that guides an agent on its course of action in a multi-agent setting (second-best approach in cases when calculation of Nash equillibra is computationally infeasible).

2. LITERATURE OVERVIEW

Traditionally, modeling of procedural rationality has aimed to represent learning processes. T. Brenner [28, Chapter 18] discusses the categorization of artificial intelligence inspired learning models including non-conscious learning, routine-based learning and belief learning. Experimental economics and psychology have made increasingly larger contributions to this knowledge in recent years, but no model has been found to be uniformly superior to others ([28, Chapters 18 and 19]).

An alternative to both the equilibrium theories (assuming substantial rationality) as well as learning approaches is constituted by the Cognitive Hierarchies (later CH) model, presented in [4][5]. The CH model consists of iterative decision rules for players doing k steps of thinking, and the frequency distribution $f(k)$ (assumed to be Poisson) of step k players. The iterative process begins with step 0 types who don't assume anything about their opponents and merely choose according to some probability distribution. Step k thinkers assume their opponents are distributed, according to a normalized Poisson distribution (with mean τ), from step 0 to step $k-1$. CH has been validated with human-based experiments where it has been found that $\tau = 1.5$ fits data from many canonical games much better than extant learning-based approaches.

Both learning-based approaches reviewed by Brenner as well as CH theory suffer from two common problems. First, they are geared towards population games, where each of the agents does not distinguish between individual opponents, but rather its payoff is derived from its own strategy and some aggregate of the population strategy. Second, the applications are most often limited to iterated single-stage (stateless) games.

The field of artificial intelligence offers solutions that do not suffer from these weaknesses. For example, [27] and [15] offer a framework for multi-agent reinforcement learning in stochastic games[6]. Given a stochastic game, the algorithms proposed converge to a Nash equilibrium when other agents are adaptable, otherwise an optimal response will be played, leading to better performance than the traditional single agent learning methods. Multi-agent reinforcement learning algorithms require agents to build internal representations of their environment and opponents, progressively updated as experience is accrued. The generality and weak assumptions of this solution come at a cost: sample complexity (measuring duration of "burn-in" phase necessary for agents to become model-consistent) is rather high, possibly too high for realistic economic modeling applications[7].

We will try to escape the trade-offs made by the extant approaches by application of recursive simulation methodology. As defined in [8], recursive simulation requires the simulated decision makers to use simulation to inform their own decision making. In our approach, the structure of internal models used by agents will be isomorphic with the structure of the model in which they themselves are embedded. Such considerations are common in the economic literature. In fact, the so called "Lucas critique" ([19]) posits that in order to design a policy intervention, one should model how individuals and institutions account (model) for the change in policy, and then aggregate the individual decisions to calculate the effects of the policy change.

Interestingly, a few applications of k-level rationality in computational models have appeared. For example, trading models [7] and [14] use it simulate a double auction, using analytically pre-calculated best responses. Recursive techniques using strategic prediction of opponent behavior have been applied to pursuit-evade tasks in fashion similar to approximation techniques used by multi-agent reinforcement learning, see [6, 29]. To our knowledge, no one has studied a case where internal models of agents are isomorphic with models in which agents are themselves embedded, structure which will be described in Section 3.

A caveat belongs here. The assumption that agents are able to reconstruct the model they belong to implies that

[4] A properly designed learning process should converge in the long run to the best response to the play by the other agents at every stage. This condition is called model-consistency and was introduced by J. Hannan in [12]. Usually, for a learning agent to become model-consistent, a long "burn-in" phase is required (it is difficult to estimate a priori it's duration, see Section 4.1). Our approach skips this phase and obtains model-consistent behaviors outright.

[5] The first review articles with empirical support for CH and related concept of k-level rationality as cognitive architecture for individuals appeared in early 2000, see [5] and [10].

[6] Stochastic game is a set of n-agent normal-form games augmented with rules for transitions depended on actions of agents. See [24, Section 2] for the definition.

[7] Y. Shoham raises this point with respect to the Trading Agent Competition ([24]).

they are representations of real-life institutions and corporations rather than individuals. Nevertheless, for application areas like mechanism design and industrial organization, this assumption fits well. In fact, it might be the case that it becomes increasingly valid as time flows, due to spread of modeling methods amongst the business community and the phenomenon of revolving doors. As noted in [13]:

> ... much regulatory rule-making is informal and case specific, and the inability of a firm to correctly forecast agency decisions imposes costs. Part of the value of of having a recent ex-regulator as an advisor will be his ability to predict - more accurately than someone without inside experience - agency decisions.

The anticipation of other players' decisions and explicitly trying to account for models used by adversaries is already part of business and regulatory activities, contributing to feasibility of using our framework as a description of oligopolistic markets[8].

3. RECURSIVE DECISION MAKING

Suppose one is given a multi agent simulation Ψ, populated with K agents. The state of the simulation at time t is defined as all relevant information, excluding policies of agents, and will be denoted as C_t. Each of the agents has an associated L-dimensional space of potential actions $A_t^i \subset \Re^L$ [9], which may depend on the current state of simulation C_t. Suppose that the behavior of agent i at time t can be described by a policy p_t^i.

The specification of policy p_t^i will be kept open and made case-specific. The set of policies for the whole population used at time t will be denoted as $\mathbf{p}_t$, a K-dimensional vector of policies.

Any agent-based simulation Ψ is a map, which for a given C_t and a fixed set of policies of agents $\mathbf{p}_t$ returns both the next state C_{t+1} as well as a vector of rewards $\mathbf{r}_t = (r_t^1, \dots, r_t^K)$ for each of the agents:

$$(\mathbf{r}_t, C_{t+1}) = \Psi(\mathbf{p}_t, C_t).$$

Two remarks need to be made about Ψ. First, in general $(\mathbf{r}_t, C_{t+1})$ are random variables, either due to nondeterministic decision rules $\mathbf{p}_t$ or to possible randomness of the agent activation scheme. Therefore, a single run of simulation Ψ yields only one realization of these variables. Second, we will assume that as long as an agent can construct description of the initial state of Ψ and assume particular policies for opponents, it can evaluate Ψ.

The ability to evaluate Ψ gives agents a powerful forecasting tool[10]. Superimposing Ψ produces forecasts about future rewards $\mathbf{r}_t, \dots, \mathbf{r}_{t+h}$ and future states of the core simulation $C_{t+1}, \dots, C_{t+h}$ for any arbitrary horizon h and scenario of policy trajectories $\mathbf{P}_{t,h} = (\mathbf{p}_t, \dots, \mathbf{p}_{t+h})$.

We will assume that agent i wants to maximize expected discounted stream of rewards for a certain planning horizon h by controlling policies $\left(p_t^i, \dots, p_{t+h}^i\right)$:

$$\max_{\left(p_t^i, \dots, p_{t+h}^i\right)} \sum_{j=0}^{h} \gamma^j E\left(r_{t+j}^i\right)$$

where γ is the discount rate[11].

For non-trivial Ψ, i's payoff and trajectory through state space will depend not only on policy agent i sets, but also on policies of the other agents.

We will assume that agents in the simulation behave according to n-th order rationality scheme. Let us denote by $\Xi^i(d,h) = (\Xi_0^i(d,h), \dots, \Xi_h^i(d,h))$ the optimal policy trajectory $(p_t^i, \dots, p_{t+h}^i)$ of i-th player in planning horizon h assuming that his order of rationality is equal to d. A 0-order rational agent replicates his last action. Assuming that the initial state of the simulation C_t and last actions of all agents $\mathbf{p}_{t-1}$ are public we get:

$$\Xi^i(0,h) = \underbrace{\left(p_{t-1}^i, \dots, p_{t-1}^i\right)}_{h+1}$$

The values of Ξ for agents having rationality order $d > 0$ are defined recursively. In each period d-th order rational player i assumes that other players $(k \neq i)$ will be playing $(d-1)$-th order rational strategy $\Xi^k(d-1,h)$. Therefore i-th player will optimize:

$$\Xi^i(d,h) \equiv \underset{\left(p_t^i, \dots, p_{t+h}^i\right)}{\operatorname{argmax}} \sum_{j=0}^{h} \gamma^j E\left(r_{t+j}^i\right)$$

subject to:

$$\forall j \in \{0, \dots, h\} : (\mathbf{r}_{t+j}, C_{t+j+1}) = \Psi(\mathbf{p}_{t+j}, C_{t+j})$$
$$\mathbf{p}_{t+j} = p_{t+j}^i \cup \{\Xi_j^k(d-1,h)\}_{k \neq i}$$

In short, the optimization process for agent i would involve following steps:

1. Clone the current state of the simulation C_t[12].

2. Assume a particular policy trajectory for other agents $\mathbf{P}_{t,h}^{-i} \equiv \mathbf{P}_{t,h} - \left\{p_t^i, \dots, p_{t+h}^i\right\}$. If $d > 0$, $\mathbf{P}_{t,h}^{-i}$ is obtained by solving problems $\Xi(d-1, \bullet)$ for competitors. If $d = 0$, by assuming that they will continue their last policies for the next h periods.

3. Adjust $\left(p_t^i, \dots, p_{t+h}^i\right)$ such that the expected discounted reward stream is maximized. The objective function is evaluated by running the core simulation Ψ for h periods forward, keeping $\mathbf{P}_{t,h}^{-i}$ fixed.

[8] One could speculate, that as more and more of economic activity is driven by models and the homogeneity of those models increases (and this includes synthetic markets with agent traders), the empirical relevance of recursive models grows

[9] Here and later we denote period number by subscript and player number by superscript

[10] We need to underline here that our focus is on domains, like industrial organization, with players already using advanced analytics to guide their decision making.

[11] Note that in order to calculate expected values of r_{t+j}^i it is necessary to run simulation Ψ multiple times.

[12] In this paper, we assume that C_t is all public information. Private information includes current policies as well as individual d and h. There is neither need for parameters d and h to be homogeneous in the population nor need they be public. In the case of heterogeneity, agent i will use his private values to instantiate $\Xi^i(d_i, h_i)$. Please see Section 6 for discussion of alternatives.

Solving $\Xi(\bullet,\bullet)$ generates an extended best-response dynamic. For $d = 1$, the best response is calculated, while $d = 2$ yields the best response to one's expectations of the others' best responses. Parameter h controls how myopic the agents are. We will discuss the consequences of setting different values of d and h in the Section 5.

4. IMPLEMENTATION AND COMPLEXITY ASPECTS

4.1 Complexity Analysis

Let's try to express computational complexity $\Xi(\bullet,\bullet)$, using as a base cost of single iteration of simulation Ψ. Four factors can influence computational complexity of the our problem:

d, K : in each recursion tree, K^{d-1} calls to optimization algorithm are made;

h, L : using a global optimization algorithm to solve decision problem of dimensionality hL, one can expect to make $O\left(e^{hL}\right)$ calls to the objective functions in the worst case.

Multiplying all the terms and using properties of big O notation, worst case computational complexity of single decision maker's $\Xi(d, h)$ is $O\left(e^{(d-1)\ln(K)hL}\right)$. This translates to the cost of whole simulation of $O\left(Ke^{(d-1)\ln(K)hL}\right)$. For comparison, numerical search for Nash equilibrium when no assumptions about Ψ can be made would cost $O\left(KhLe^{KhL}\right)$, see [23]. This means that even for large d, the worst case computational cost of d-th order rationality is significantly smaller than of substantive rationality.

Furthermore, there are at least two ways of reducing this cost. First, note that we have assumed that state of simulation C_t is public. In such a case, for $d \geq 2$, decision makers on the market will be solving a lot of identical subproblems that can be pruned to reduce the complexity of the whole simulation to $O\left(Ke^{hL} + e^{(d-2)\ln(K)hL}\right)$.

Second, agent's can trace how such surfaces and decisions they imply change with each additional degree of rationality, ceteris paribus. In particular, we can define a marginal change to policy with an additional degree of rationality as:

$$\delta(d, C_t) = \left\|\Xi^i(d+1, 1)\left(C_t\right) - \Xi^i(d, 1)\left(C_t\right)\right\|$$

and then sample from the space of possible set of simulation states to establish the shape of $\mathrm{E}\left(\delta(d)\right)$. As discussed in [5], under k-th order rationality, as k grows large, players doing k and $k+1$ steps of thinking will, in the limit, have the same beliefs, make the same ex ante choices, and have the same expected payoffs, therefore we would expect $\lim_{d\to\infty} \mathrm{E}\left(\delta(d)\right) = 0$ for our problem as well. This observation opens a possibly of agents doing a dynamic cost-benefit analysis on parameters d, h, endogenizing rationality levels when computation is costly.

4.2 Implementation

Figure 1 presents a general design pattern for recursive simulations. The key part of the design includes making the simulation object implement `Cloneable` and `Probeable` interfaces[13], such that each strategic agent can instantiate a self contained simulation, one necessary to solve the $\Xi(\bullet,\bullet)$ optimization problem.

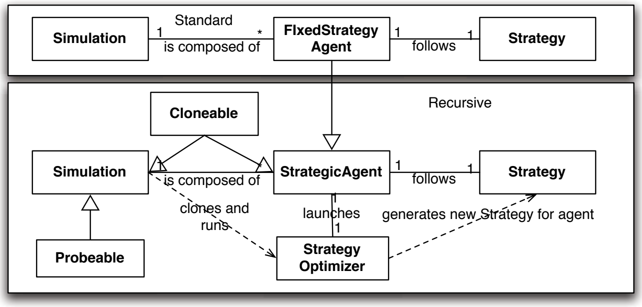


Figure 1: A canonical agent-based simulation (upper panel) augmented with recursive agents (lower panel). The simulation object needs to be queried for payoffs and policy profiles, an ability ensured by implementing a `Probeable` interface. Both simulation as well as all agents need to be `Cloneable`. Independence of external resources helps to keep it self contained. `StrategicAgent`s need to be coupled with a generic global optimization solver used to search the `policy` space. `StrategyOptimizer` uses a cloned simulation to evaluate the fitness of policies and the solve $\Xi\left(\bullet,\bullet\right)$ problem.

As many layers of simulations contained within simulations might need to be instantiated, the framework used needs to be relatively lightweight. Note that $\Xi(\bullet,\bullet)$, thanks to the tree-like nature of each of the decision problems, could be parallelized easily in multithreaded environments. As long as depth of recursion d is large enough and agents are responding simultaneously to the same information set C_t, distributed implementation should offer full benefits of scale. For this purpose, we have adopted the discrete simulation engine described in [20].

Of great importance is the issue of the optimization algorithm applied. Examples from Section 5 were run using the real valued genetic algorithm provided by off-the-shelf library OAToolbox developed by [3]. In addition, the framework has been integrated with the Gamut library ([22]) and all of the games implemented there might be played with our n-th order rational agents[14].

5. VERIFICATION

For the purpose of verifying the correctness of our implementation, we will use a non-stationary iterated game. Suppose one is given a bimatrix game B and uses it to define a Φ mapping with properties from Section 3. We will start by listing components of B:

[13] `Cloneable` interface provides a deep copy of Java objects, while the `Probeable` interface is our custom interface providing probes returning accrued payoffs and strategies of agents present in the simulation.

[14] Materials pertaining to our simulation can be downloaded from `https://www.assembla.com/wiki/show/recursiveengines`. In particular, the website features more experiments establishing face and technical validity of the framework, docking it with extant models and checking the robustness of the obtained results. Working papers describing two applications ([18, 16]) can also be accessed at that location.

$$\mathrm{B} = \left\langle A^1, A^2, Q^1\left(A^1, A^2\right), Q^2\left(A^2, A^1\right)\right\rangle$$

where A^i and Q^i are the action-set and payoff matrix for agent i. Assume that behavior of player i at time t is described with probability distribution p_t^i over set A^i. In such a case, the single-stage payoff for player i, $\mathrm{B}^i\left(p_t^i, p_t^{-i}\right)$ can be obtained by simple multiplication:

$$\mathrm{B}^i\left(p_t^i, p_t^{-i}\right) = p_t^i Q^i p_t^{-i}$$

Game B is too simple too make a good verification testbed. We will inject the path dependency into game by making policy adjustment costly and define a new game Φ. Suppose that player i using policy p_t^i has determined that in face of the policy of opponent p_t^{-i}, it is desirable to change its policy to p_{t+1}^i. We can modify i's payoff using following transformation of stage payoff:

$$\Phi^{1,i}\left(p_t^i, p_t^{-i}, p_{t+1}^i\right) = p_t^i Q^i p_t^{-i} - \delta\left\|p_t^i - p_{t+1}^i\right\|$$

The information set used at time t will include past policies and the stage payoff matrix, $C_t = \left\{p_{t-1}^1, p_{t-1}^2, \mathrm{B}\right\}$. This yields the second component $\Phi^{2,i}$ in a natural way.

We will use this trick to create two testbeds, one based on the generalized Shapley bimatrix game, the other on the so called Colonel Blotto bimatrix game. The Shapley game will be used to dock $\Xi(\bullet,\bullet)$ with low-rationality behaviors considered in [26] and look for evidence of policy convergence with increase in d. The Colonel Blotto game will investigate differential advantages agents can obtain by using $\Xi(\bullet,\bullet)$ logic and compare simulated policies with results of [9]. In addition, computational cost of $\Xi(\bullet,\bullet)$ will be described.

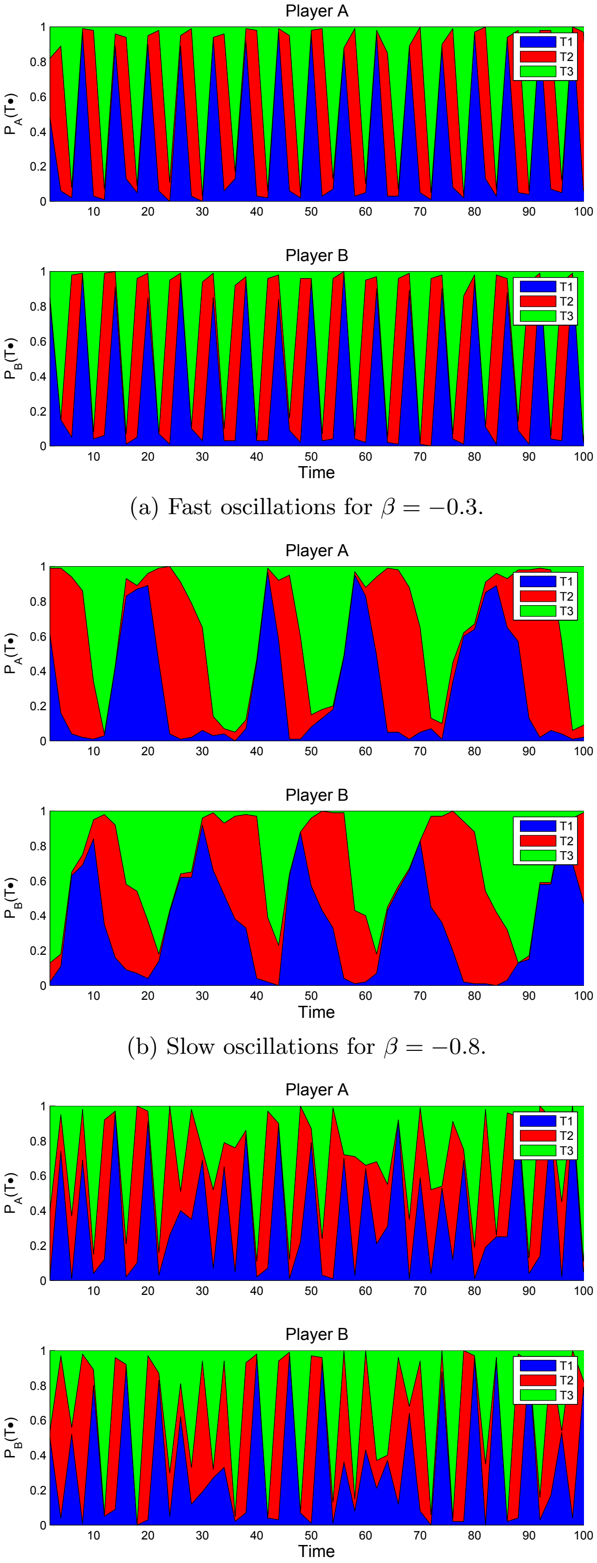
5.1 Generalized Shapley Game

In the 1960s Shapley provided an example of a two-player fictitious game with periodic behavior. In this game, agent A aims to copy agent's B behavior and agent A aims to play one ahead of agent B. In [26] Shapley's example was generalized by introducing an external parameter β. Two agents A and B play following family of 3×3 games with single stage payoffs determined by the matrices:

$$A = \begin{pmatrix} 1 & 0 & \beta \\ \beta & 1 & 0 \\ 0 & \beta & 1 \end{pmatrix} \quad B = \begin{pmatrix} -\beta & 1 & 0 \\ 0 & -\beta & 1 \\ 1 & 0 & -\beta \end{pmatrix}$$

It has been shown the periodic behavior in Shapley's example at some critical value of β disintegrates into unpredictable (chaotic) behavior, with players dithering a huge number of times between different policies. At a further critical parameter the dynamics becomes periodic again, but now both players aim to play one ahead of the other. As proved in [26] those two critical parameters are σ equal to the golden mean ≈ 0.618 and $\tau \approx 0.915$ such that:

- for $\beta \in (-1, \sigma)$ the players typically end up repeating Shapley's pattern; in fact, for $\beta \in (-1, 0)$ regardless of the initial positions, the behavior tends to a periodic one (for $\beta \geq 0$ this is not true: then there are orbits which tend to the interior equilibrium and/or other periodic orbits).



(a) Fast oscillations for $\beta = -0.3$.

(b) Slow oscillations for $\beta = -0.8$.

(c) Erratic (possibly chaotic) behavior for $\beta = 0.8$.

Figure 2: Evolution of mixed policies p_t^A and p_t^B for three generalized Shapley games with different values of β. The parameter $\delta = 0.5$ for all of the simulations and behavioral logic used was $\Xi(1,1)$.

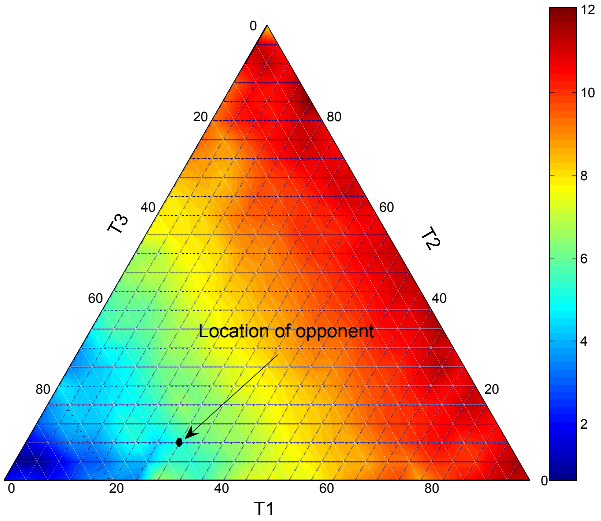


Figure 3: Sample policy landscape for Blotto game under $\Xi(3, 1)$ policy with $\delta = 0$. Fitness (color-coded z-axis) measures total payoff from 100 Monte Carlo samples for each p_t^i. Note that in this case, no point global maximum exists and any p_t^i where probability of taking T2 is assigned 0 has similar payoff. For this particular game, lack of unique global maximum in solution of $\Xi(\bullet, \bullet)$ may prevent convergence of policies as the degree of rationality grows (see Table 1 for more details).

- for $\beta \in (\sigma, \tau)$ the players become extremely indecisive and erratic (and the moves become chaotic), while
- for $\beta \in (\tau, 1)$ the players typically end up playing the anti-Shapley pattern.

We posit that there exists δ, such that two players using $\Xi(1, 1)$ replicate dynamics presented in [26][15]. Figure 2 features three sample runs of generalized Shapley game for different values of β with $\delta = 0.5$. Shapes of policy trajectories are consistent with those predicted by apparatus of [26], showing that under costly adjustment of strategies, for $d = 1$ our framework would generate similar answers to dynamics caused by fictitious best responses.

In the next section we investigate what happens when level of rationality d is increased.

5.2 Colonel Blotto Game

Colonel Blotto game is a zero-sum game of strategic mismatch between two agents X and Y, studied by E. Borel and first solved in [11]. A policy p_t^x for player X can be written as a real vector $p_t^x = \{x_t^1, x_t^2, \ldots, x_t^m\}$ with $\sum_{i=1}^m x_t^i = 1$, $x_t^i \in [0, 1]$, where x_i represents the fraction of the budget allocated to front i, and m is the total number of fronts. Here, both players are symmetric and have the same available budget. The single-stage payoff to p_t^x against p_t^y is:

$$\sum_{i=1}^{m} \operatorname{sgn}\left(x_t^i - y_t^i\right)$$

where the function

$$\operatorname{sgn}(\xi) = \begin{cases} -1 & \text{if } \xi < 0 \\ 0 & \text{if } \xi = 0 \\ 1 & \text{if } \xi > 0. \end{cases}$$

We assume $m > 2$. Otherwise, the game always ends in a tie. It can be shown that there is no pure strategy Nash Equilibrium. A pure strategy p_t^x that allocates resources to the i-th front, $x_t^i > 0$, will lose to the strategy p_t^y that allocates no resources to the ith front and more resources to all other fronts:

$$y_t^i = 0,\ y_t^j = x_t^j + \frac{x_t^i}{m-1}\ \forall j \neq i$$

[11] showed that the Colonel Blotto game has a mixed strategy equilibrium in which the marginal distributions are uniform on $[0, \frac{2}{m}]$ along all fronts. This unpredictability leaves the opponent with no preference for one strategy or another as long as no front is allocated more than $\frac{2}{m}$ resources. More modern treatment of this game can be found in [9], where interactions between individual fronts were permitted.

Colonel Blotto game is used to test influence of rationality levels on payoffs of agents. First, please refer to Figure 4 for a number of sample game trajectories under different conditions. On that Figure, the normalizing influence of δ is very explicit. When costs are significant, the main strategy applied by player is to try to match the distribution of opponents and then try to shift minimal possible increments such that victory is ensured (this corresponds to trajectories that first intersect, and then turn into small scattered clouds around the intersection point). Second, Table 1 summarizes performance of two players, d_r-th order rational row player and d_c-th order rational column player in Colonel Blotto game:

1. Symmetric $d_c = d_r$ leads to the same payoffs for both players (equal to 0 as the game is zero-sum). This holds only in the long run. In the short run, payoff variance may be significant;
2. The maximum payoff row player obtains happens when his decision making model is true (that is, when column player is of type $d_c = d_r - 1$), regardless of *delta*.
3. For fixed rationality of column player and low adjustment cost ($\delta = 0$), increasing d_r when $d_r > d_c+1$ leads to decrease in payoff (cost of over strategizing).
4. On average, payoffs decrease as rationality levels increase and approach 0. Also, relative payoff's variance decreases in the symmetric case (which means players start playing equilibrium strategies);

Summarizing, for large k, there is no marginal reward for a k-th order player to think harder (as Table 1 shows, in

[15] If penalty $\delta = 0$, past policies do not constrain players, which start playing equilibrium mix ($\{\frac{1}{3}, \frac{1}{3}, \frac{1}{3}\}$ for both A as well as B) as d increases. Setting $\delta > 0$ and $\Xi(1, 1)$ is equivalent to fictitious play assumption [26], where the strategies of opponents, against which best responses are calculated, are updated gradually.

		Game for $\delta = 0$						Game for $\delta = 0.5$					
		Rationality of column player						Rationality of column player					
		0	1	2	3	4	5	0	1	2	3	4	5
Rationality of row player	0	0	−99	−5	−4	−1	0	0	−99	−98	−92	−93	−91
	1	99	0	−23	0	3	4	99	0	−47	−48	−53	−47
	2	5	23	0	−10	−7	−2	98	47	0	−29	−40	−45
	3	4	0	10	0	−5	4	99	48	29	0	−19	−25
	4	1	−4	7	5	0	−4	99	53	40	19	0	−12
	5	−1	−2	2	−4	4	0	92	47	45	25	12	0

Table 1: Summary of performance of players of different levels of rationality in the Colonel Blotto game with $\delta \in \{0, 0.5\}$. As the game is zero-sum, only payoffs of the row player were recorded. The payoffs are scaled as percents of the maximal theoretical payoff, averaged out over 1000 runs per setting, duration of each was set to 100 iterations. At 0.05 significance level, green payoffs are significantly > 0, red are less than < 0 and blue can not be distinguished from 0.

context of Blotto game, it may be even detrimental if adjustment costs are small). We posit that this is caused by the lack of the unique solution to $\Xi(\bullet, \bullet)$, situation visualized on Figure 3. This means that for some games, limits to strategic thinking exist and that those limits would apply to any form of rationality, procedural or substantive (called equilibrium selection problem in that specific context). The benefit of our approach is that we can explicitly quantify for any situation how far one can go before reasoning stops benefiting decision maker. This opens up a interesting research question of designing cost-benefit analysis scheme that would account for that danger and dynamically adjust rationality levels.

6. SUMMARY AND EXTENSIONS

In this paper, we have introduced a context-independent computational implementation of n-th order rationality and demonstrated its functionality on two test cases. We showed how an n-th order rationality model deviates systematically from equilibrium predictions as agents are engaged in a multi-tiered game of outguessing each others' responses to the current state of world. We presented a structural design that enables introduction of n-th order rational agents into any agent-based model and demonstrated that agents using n-th order rationality are model-consistent and do not require learning in order to behave well.

We have demonstrated the ease with which sensitivity analysis with respect to rationality assumptions can be performed. While doing so, we identified a number of open questions which will serve as a base for future research. First, we are performing more disciplined investigation of sensitivity / sensibility of $\Xi(\bullet, \bullet)$ with respect to d and h for wide classes of games. Second, we are trying to establish a link between existence of non-unique solutions to $\Xi(\bullet, \bullet)$ and equilibrium selection problem from game theory / prediction boundaries problem. Lastly, we are intend to develop analytical convergence and regret proofs for $\Xi(\bullet, \bullet)$ docking it within other results from game theory and machine learning.

Concurrently with theory work, we found the framework to be extensible with respect to the perturbation of definitions of strategies and the structure of underlying Ψ. For example, in [16], we propose a computational model of a telecommunication market with realistic call patterns among customers. In that paper, a market design question is tackled: identification of a robust plan for regulating interconnection fees. The strategy space includes multiple price trajectories that need to be set by n-th order rational operators. Additionally, that model is filled with thousands of non-strategic customers that used simple heuristics to provide operators with feedback on market effects of their decisions.

Another angle is investigated in [18], a simulation of irregular warfare environments. In that paper, we study co-evolution of strategies between security and terrorist organizations, allowing for assymetric information about state of the simulation C. All codes, supplementary results as well as application papers can be downloaded from `https://www.assembla.com/wiki/show/recursiveengines`.

Acknowledgments

First author was supported in part by ONR Basic Research grant #00332415. All authors would like to acknowledge input of participants of seminars at George Mason University and Warsaw School of Economics.

7. REFERENCES

[1] R. Axtell. The Complexity of Exchange. *The Economic Journal*, 2005.

[2] K. Binmore. Modeling rational players i. *Economics and Philosophy*, 3:179–214, 1988.

[3] J. Brownlee. Oat: The optimization algorithm toolkit. Technical report, Complex Intelligent Systems Laboratory, Swinburne University of Technology, 2007.

[4] C. F. Camerer, T.-H. Ho, and J.-K. Chong. A cognitive hierarchy model of games. *The Quarterly Journal of Economics*, 119(3):861–898, August 2004.

[5] V. P. Crawford and N. Iriberri. Level-k auctions: Can a nonequilibrium model of strategic thinking explain the winner's curse and overbidding in private-value auctions? *Econometrics*, 75:1721–1770, 2007.

[6] E. H. Durfee and J. Vidal. Recursive agent modeling using limited rationality. *Proceedings Of The First International Conference On Multi-Agent Systems*, pages 125–132, 1995.

[7] E. H. Durfee and J. Vidal. Building agent models in economic societies of agents. *AAAI-96 Workshop on Agent Modeling*, pages 90–97, 1996.

[8] J. Gilmer and F. Sullivan. The use of recursive simulation to support decisionmaking. In *Proceedings of the 2003 Winter Simulation Conference*, 2003.

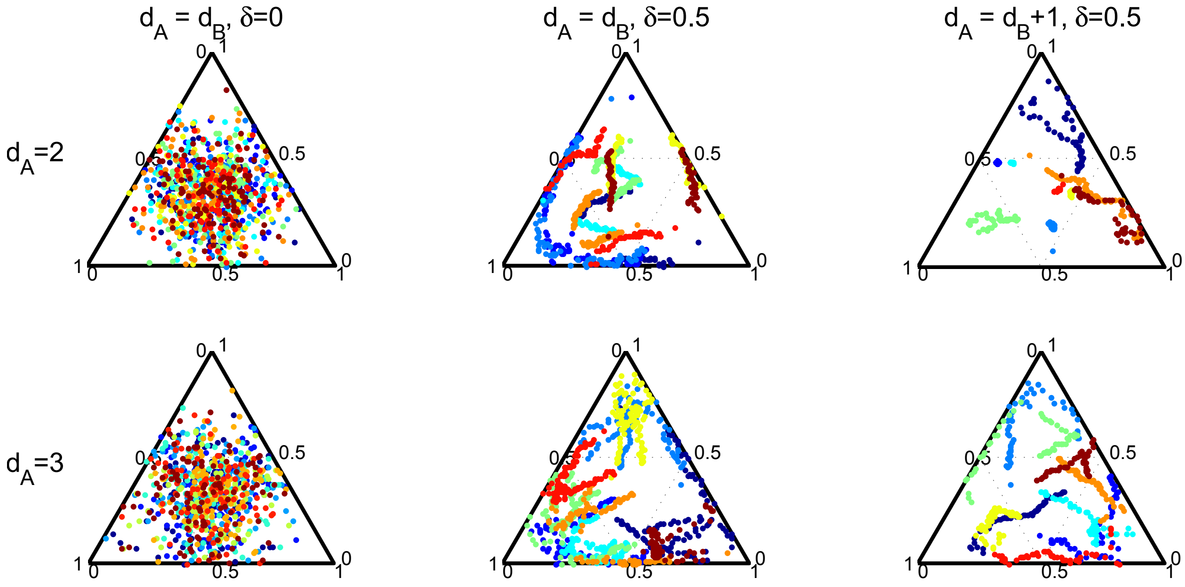


Figure 4: Sample trajectories for Blotto game under different rationality levels and choices of δ. For each setting, 50 periods for 10 games are plotted, with trajectories of both player marked with the same color.

[9] R. Golman and S. Page. General blotto: games of allocative strategic mismatch. *Public Choice*, 2008.

[10] M. Gomes and V. Crawford. Cognition and behavior in normal-form games: An experimental study. *Econometrica*, 69:1193–1235, 2001.

[11] O. Gross and R. Wagner. *A Continuous Colonel Blotto Game*. Rand Corporation, 1950.

[12] J. Hannan. Approximation to Bayes Risk in Repeated Plays. *Contributions to the Theory of Games*, 3, 1959.

[13] A. Heyes. Revolving doors and regulatory complexity. Royal Holloway, University of London: Discussion Papers in Economics 99/1, Department of Economics, Royal Holloway University of London, Feb. 2000.

[14] J. Hu and M. P. Weliman. Learning about other agents in a dynamic multiagent system. *Cognitive Systems Research*, 2:67–79, 2001.

[15] J. Hu and Y. Zhang. Online reinforcement learning in multiagent systems. In *Proceedings of 8th Conference of American Association for Artificial Intelligence*, 2002.

[16] B. Kaminski and M. Latek. The influence of call graph topology on the dynamics of telecommunication markets. Technical report, 2008.

[17] J. M. Keynes. *The General Theory of Employment, Interest and Money*. Macmillan Cambridge University Press, 1935.

[18] M. Latek and M. Tsvetovat. Strategic interaction of forces in irregular warfare: Agent-based model. Technical report, Center for Social Complexity, George Mason University, 2008.

[19] R. Lucas. Econometric Policy Evaluation: A Critique. *Carnegie-Rochester Conference Series on Public Policy*, 1976.

[20] S. Luke, C. Cioffi-Revilla, L. Panait, K. Sullivan, and G. C. Balan. Mason: A multi-agent simulation environment. *Simulation*, 81(7):517–527, 2005.

[21] K. Michihiro. *Advances in Economics and Econometrics: Theory and Applications, Seventh World Congress*, chapter Evolutionary Game Theory in Economics, pages 244–279. Cambridge University Press, 1997.

[22] E. Nudelman, J.Wortman, Y. Shoham, and K. Leyton-Brown. Run the gamut: A comprehensive approach to evaluating game-theoretic algorithms. In *Proceedings of the Third International Joint Conference on Autonomous Agents and Multiagent Systems*, 2004.

[23] C. Papadimitriou. *Computational Complexity*. Addison-Wesley, 1994.

[24] Y. Shoham, R. Powers, and T. Grenager. Multi-agent reinforcement learning: a critical survey. In *AAAI Fall Symposium on Artificial Multi-Agent Learning*, 2004.

[25] H. Simon. *The Sciences Of The Artificial*. The MIT Press, 1982.

[26] C. Sparrow, S. van Strien, and C. Harris. Fictitious play in 3x3 games: The transition between periodic and chaotic behaviour. *Games and Economic Behavior*, 63(1):259–291, May 2008.

[27] N. Suematsu and A. Hayashi. A multiagent reinforcement learning algorithm using extended optimal response. In *Proc. of 1sr Int. Conf. on Autonomous Agents and Multiagent Systems (AAMAS 2002)*, 2002.

[28] L. S. Tesfatsion and K. L. Judd. *Handbook of Computational Economics, Vol. 2: Agent-Based Computational Economics*. Elsevier, May 2006.

[29] J. M. Vidal and E. H. Durfee. Recursive agent modeling using limited rationality. In *Proceedings of the First International Conference on Multi-Agent Systems*, 1995.