The Knowledge Engineering Review, Vol. 27:2, 239–254. © Cambridge University Press, 2012
doi:10.1017/S026988891200015X

Individual evolutionary learning with many agents

JASMINA ARIFOVIC[1] and JOHN LEDYARD[2]

[1]*Department of Economics, Simon Fraser University, 8888 University Drive, Burnaby, BC V5A 1S6, Canada;*
e-mail: arifovic@sfu.ca;
[2]*Division of Humanities and Social Sciences, California Institute of Technology, 1200 East California Boulevard, MC 228-77, Pasadena, CA 91125, USA;*
e-mail: jledyard@hss.caltech.edu

Abstract

Individual Evolutionary Learning (IEL) is a learning model based on the evolution of a population of strategies of an individual agent. In prior work, IEL has been shown to be consistent with the behavior of human subjects in games with a small number of agents. In this paper, we examine the performance of IEL in games with many agents. We find IEL to be robust to this type of scaling. With the appropriate linear adjustment of the mechanism parameter, the convergence behavior of IEL in games induced by Groves–Ledyard mechanisms in quadratic environments is independent of the number of participants.

1 Introduction

In Arifovic and Ledyard (2011), we study Individual Evolutionary Learning (IEL), a behavioral learning model that is applicable to repeated games with large strategy spaces, including the continuum. In that paper, we used IEL in the games generated by Groves–Ledyard (GL) mechanisms in a quadratic environment. The GL mechanism solves the free-rider problem for public goods. Agents send messages that determine the size of the public good. Agents are taxed, based on the messages they send and on the difference between their message and the average of the other agents' messages. The mechanism is balanced and efficient. That is, the mechanism collects exactly the taxes needed to pay for the amount of the public good produced and, at the Nash Equilibrium messages, the allocation implied by the public good choice and the taxes is Pareto-optimal. One distinctive feature of the GL mechanisms is a punishment parameter, γ. As this parameter is increased, it creates stronger incentives for each agent to match the average message of the other agents. The particular value of γ does not affect the static properties of the mechanism. It is balanced and efficient for all positive values of γ.

However, dynamic properties, such as the time to converge to Nash Equilibrium in repeated play, do depend on the particular value. Time to converge is very important if these mechanisms are to be used in practice. The welfare (utility or payoff) of the agents is higher the closer to Pareto-optimal allocations they are. Since Nash Equilibria are Pareto-optimal, the faster the agents converge to Nash Equilibrium, the better off they will be over time. Standard theories of dynamics, based on best-reply dynamics and strategic complementarities, suggest that, for these games, there is a number such that if γ is larger than that number then convergence occurs and if γ is smaller then convergence will not occur. Contrary to these standard theories, the prediction from the IEL behavioral model is that the average time to convergence varies smoothly and is U-shaped in γ. We validated these predictions with data from economic experiments with human subjects. However, all of the modeling and experiments were done with five players. Real economies involve many more than that.

https://doi.org/10.1017/S026988891200015X Published online by Cambridge University Press

In this paper we investigate what happens when the number of players increases. One can imagine many possible conjectures about the effect of numbers on the learning behavior of agents operating in a repeated game context. For example, it is possible that with boundedly rational agents who make mistakes, as occurs in IEL, an increase in numbers could lead to more errors and more erratic behavior by compounding the errors. Alternatively, an increase in numbers could, through the law of large numbers, reduce the variability seen by any one agent, which would enable them to be closer to fully rational behavior.

In this paper we show that if the increase in agents occurs through replication and if the mechanism parameter is normalized for the population size, then the convergence properties of IEL do not depend on the size of the population. That is, let γ be the mechanism parameter, N be the number of agents, and T_c^γ be the time of convergence[1]. Then $T_c^\gamma = f(\gamma/N)$, so that if strategies converge to stage game equilibria in 10 rounds when $N = 5$ and $\gamma = 50$, then strategies will converge in 10 rounds when $N = 100$ and $\gamma = 1000$. Interestingly, the size of the economy does not seem to affect the rate of behavioral learning by IEL.

1.1 Environments, mechanisms, and games

Public-good environments. There are N agents, $i \in \{1, \ldots, N\}$. Let $z = (X, y^1, \ldots, y^N)$, where X is the amount of the public good and y^i is i's net consumption of the private good. Each agent i begins with an initial endowment of the private good α^i. Agent i's consumption of the private good is $\alpha^i + y^i$. The per-person cost of production for one unit of the public good is c. The set of feasible allocations is

$$Z = \left\{ z = (X, y) | X \geqslant 0, NcX + \sum_{i=1}^{N} y^i = 0 \right\}$$

Each consumer i has a utility function: $V^i(X, \alpha^i + y^i) = A^i X - B^i X^2 + \alpha^i + y^i$.

We use the notation (N, Z, V) to denote an environment. One allocation of interest in an environment is the one that maximizes the sum of the utilities subject to feasibility. It is easy to see that the X that does this is

$$\hat{X} = \frac{\left(\sum_i A^i\right) - Nc}{2\sum_i B^i}$$

Groves–Ledyard Mechanisms[2]. Begin with a language, a space of messages, $M = (-\infty, +\infty)$ and $m^i \in M$ can be thought of as i's incremental demand for the public good.

Given a vector of messages $m = (m^1, \ldots, m^N)$, the public good produced is

$$X(m) = \sum_{i=1}^{N} m^i$$

The tax to be paid by i is

$$T^i(m, \gamma) = X(m)(c/N) + (\gamma/2)\left[\frac{N-1}{N}\left(m^i - \mu_{-i}\right)^2 - \sigma_{-i}^2\right]$$

where γ is an arbitrary free parameter greater than 0, $\mu_{-i} = \sum_{h \neq i} m^h/(N-1)$ is the mean value of the messages of the other agents, and $\sigma_{-i}^2 = \frac{\sum_{h \neq i}(m^h - \mu_{-i})^2}{N-2}$ is the squared deviation from this mean. A GL mechanism is $(M, g(m,\gamma))$, where $g(m, \gamma) = (X(m), y^1(m, \gamma), \ldots, y^N(m, \gamma)) = (X(m), -T^1(m, \gamma), \ldots, -T^N(m, \gamma))$.

[1] These will be defined more precisely later.

[2] This section is intended mainly as a reminder to the reader of the formal structure of the problem. For more details, see Groves and Ledyard (1977) or Chen and Plott (1996).

https://doi.org/10.1017/S026988891200015X Published online by Cambridge University Press

Different values of γ imply different outcome functions and, therefore, different mechanisms. So by letting γ range over values in $(0, \infty)$, one creates an entire class of mechanisms.

Games. An environment and a mechanism combine to create a game, $G = \{N, H, u\}$, where N is the number of players, a strategy of i is $h^i \in H^i$ the strategy set of player i, and $u^i(h)$ is the payoff to i if players use h. For the GL mechanism, $(M, g(m,\gamma))$, in the public-good environment, (N, Z, V), combining is straightforward. The N players in the game are the N agents in the environment. The strategy set of player i, $H^i = M^i$. Let $r^i(m) = (\mu_{-i}(m), \sigma^2_{-i}(m))$. The payoff to i is $u^i(m) = W^i(m^i|r^i(m), \gamma)$ where, because r^i does not depend on m^i and $X(m) = m^i + (N-1)\mu_{-i}$,

$$W^i(m^i|r^i(m), \gamma) = V^i(m^i + (N-1)r^i_1(m), \alpha^i - T^i(m^i, r^i(m), \gamma)) \tag{1}$$

We let $G(\gamma)$ be the game derived this way. As γ ranges over $(0, \infty)$, a continuum of games is generated. The Nash equilibrium strategy for i in the game $G(\gamma)$ is $h^i(\gamma)$ where

$$\hat{h}^i(\gamma) = \frac{\hat{X}}{N} + \frac{A^i - 2B^i\hat{X} - c}{\gamma} \tag{2}$$

Growing the economy. In this paper we are concerned with what happens as we increase the size of the economy. To keep things comparable as N grows, we consider economies in which larger economies contain replicates of the smaller economies. Let $E = (N, Z, V)$ be an economy. The k-th replicate of E will have kN individuals. If V^i is the utility of one person in E, there will be k people in the k-th replicate with that utility function. We let $E(k) = (N(k), V(k), Z(k))$ represent the k-th replicate of E.

The level of public good that maximizes the sum of utilities in $E(k)$ is

$$\hat{X}(k) = \frac{k\left(\sum_i A^i\right) - kNc}{2k\sum_i B^i} = \frac{\left(\sum_i A^i\right) - Nc}{2\sum_i B^i} = \hat{X}$$

That is, the optimal level of public good does not change as we increase the size of the economy. The reason is that we have assumed that the per-capita cost of a unit of the public good is the same in all replicates. If, for example, there was a constant marginal cost for the public good, independent of the size of the economy, then as the economy grew and the aggregate marginal benefit of the public good increased, the optimal level of the public good would increase. With a constant per-capita marginal cost, the marginal per-capita benefit and per-capita cost do not change as the economy grows and so the optimal level of public good does not change.

The equilibrium strategy for i in the game generated by $E(k)$ is

$$\hat{h}^i(\gamma) = \frac{\hat{X}}{kN} + \frac{A^i - 2B^i\hat{X} - c}{\gamma} \tag{3}$$

We will see below that there will be a reason to normalize γ by the size of the economy. That is, we will want to let $\gamma = k\hat{\gamma}$ as the economy grows. In this case, the equilibrium strategy for i in the game generated by $E(k)$ is

$$\hat{h}^i(\gamma) = (1/k)\left\{\frac{\hat{X}}{N} + \frac{A^i - 2B^i\hat{X} - c}{\hat{\gamma}}\right\} \tag{4}$$

so that $\hat{h}^i(\gamma) \to 0$ as $k \to \infty$.

1.2 Prior results

The theoretical equilibrium properties of GL mechanisms are well understood. The tax and allocation rules are specifically designed so that, if the agents follow Nash equilibrium behavior, then the equilibrium outcome of a one-shot game will be a Pareto-optimal allocation. Formally, from (2), if $\hat{m}$ is a Nash equilibrium of $G(\gamma)$ then $\sum_i \hat{m}^i = \hat{X}$. In environments with quasi-linear preferences, the Pareto-optimal level of public good is unique and the equilibrium outcome level of

https://doi.org/10.1017/S026988891200015X Published online by Cambridge University Press

the public good is independent of γ. But if one is interested in actually using these mechanisms, it is necessary to understand their dynamics. For example, if one is interested in the ability of the mechanisms to attain optimal levels of utility in repeated situations, then one must ask whether and how fast individuals will converge to the Nash equilibrium, since faster convergence implies higher aggregate welfare.

Theory is mostly silent on the dynamics of GL mechanisms. Three exceptions are papers by Chen and Tang (1998), Muench and Walker (1983), and Page and Tassier (2004), all of which suggest that the parameter γ plays a major role in those dynamics for agents following adaptive strategies. Based on the work of Milgrom and Roberts (1990) on strategic complementarities, Chen and Tang (1998) derive a sufficient condition for quadratic preferences, $\gamma/N \geqslant 2B^i$ for all i, for the convergence of the mechanism in a sequence of repeated one-shot games if agents use adaptive learning[3]. Another sufficient condition for global convergence to Nash equilibrium, if agents use best response, can be derived from a theorem of Gabay and Moulin (1980) using a dominant-diagonal condition[4]. For quadratic preferences that condition holds if $\gamma/N \geqslant [(N-2)/(N-1)]B^i$ for all i. However, neither the strategic complementarity nor the dominant-diagonal condition provide any insight into how the speed of convergence might depend on γ. Such knowledge is particularly important for practical implementations.

Muench and Walker (1983) examined the dynamics of GL mechanisms in large economies under best-response behavior. They found that if γ were fixed as N grew, then there was a $\hat{k}$ such that for $k > \hat{k}$ the dynamics were unstable[5]. They showed that this instability could be avoided if γ were allowed to grow with the economy. Let $\gamma(k) = kN\hat{\gamma}$. Then best-response dynamics are stable for our environments, as can be seen in the sufficient condition of Gabay–Moulin. If $\hat{\gamma} \geqslant \max_i B^i$, then $\gamma(k)/N(k) = \hat{\gamma} \geqslant (kN-2))/(kN-1)\max\ B^i$, for all $k>0$. But they then pointed out that this would cause the utility of each agent to flatten out in the sense that the utility they get from the equilibrium best response is not much different than the utility they get from using the strategy[6] $m^i = \mu_{-i}$. So if there are any cognitive costs, the agents will have little incentive to move to the optimal equilibrium.

Page and Tassier (2004) report on a number of simulations with Q-learning[7] in the Chen-Tang (1998) environments. For the Chen-Tang parameters, the strategic-complements condition for convergence is that $\gamma \geqslant 80$ and the Gabay–Moulin condition is that $\gamma \geqslant 30$. In Q-learning, agents respond with a weighted average of q on their last period message and $(1-q)$ on their best response. Interestingly Page and Tassier report (2004: 318): 'For γ greater than 12 the agents converge for any level of q'. That is, they also find convergence for a wider range of γ than covered by the sufficient conditions. They also report that 'For $\gamma = 1$, q needs to be greater than ~ 0.55 to ensure convergence'. But neither their paper nor any of the others provides any guidance as to the rate of convergence.

In Arifovic and Ledyard (2011), our goal was to identify a behavioral learning model that was consistent with the behavior observed in the economic experiments with human subjects. We used two sets of experimental results from the GL environments: a set reported by Chen and Tang (1998), and a set that we generated in July 2007. In our experiments, we followed Chen and Tang's design. Thus, in the simulations and the analysis of our behavioral model, we only considered

[3] Adaptive learning is defined in Milgrom and Roberts (1990) and includes best response, fictitious play, Bayesian Learning, and others. The sufficient condition for convergence under adaptive learning is $\partial^2 V^i/\partial m^i \partial m^j \geqslant 0$.

[4] We thank Paul Healy for the Gabay–Moulin reference. See Healy (2006) for a use of the theorem in the context of public good mechanism design. The diagonal condition is satisfied if $|\partial^2 W^i/\partial m^i m^i| > \sum_{i \neq j} |\partial^2 W^i/\partial m^i m^j|$.

[5] This follows from the Gabay–Moulin condition since the left-hand side of the inequality goes to zero as $N \to 0$, while the right-hand side is bounded away from zero.

[6] It can be shown that $lim_{k\to\infty}\, u^i(\hat{m}(kN\hat{\gamma})) - u^i(\hat{m}(kN\hat{\gamma})/\mu_{-i}) = 0$.

[7] See Watkins (1989).

https://doi.org/10.1017/S026988891200015X Published online by Cambridge University Press

groups of size $N = 5$. In our simulations, IEL exhibited behavior consistent with humans but at odds with the theoretical predictions. There were three main findings. (1) There is convergence to Nash equilibrium messages of the stage game for all of the values of γ that we simulated, including $\gamma = 1$ and convergence is fast for a much larger set of the values of γ than that predicted by either the strategic-complementarity condition or the dominant-diagonal condition. (2) The time to first convergence is smooth and U-shaped in γ with the minimum average convergence time occurring at around $\gamma = 50$. (3) Convergence is stable in the sense that once the model first nears the equilibrium, it remains in its neighborhood.

This of course leaves open the question as to whether these findings with IEL would survive scaling up to larger groups. We turn to that now.

2 Description of the learning algorithm

Our algorithm, Individual Evolutionary Learning (IEL) is based on the evolutionary paradigm that successful strategies thrive and increase in frequency over time and that there is occasional experimentation. This paradigm has been usually associated with models of social learning, where agents learn by imitating and adopting strategies of more successful agents[8]. However, in IEL the evolution of strategies takes place at the level of the individual agent, who has a collection of remembered strategies that is updated over time[9].

2.1 Individual evolutionary learning

An environment and a mechanism combine to form a game which can be turned into a repeated game. The repeated game has a stage game G and a number of rounds, T. The idea is that G will be played for T rounds. In $G = \{N, H, u, r\}$, N is the number of subjects, H^i is the strategy space of i, $u^i(h^i, \ldots, h^N)$ is i's payoff if the joint strategy choice is h, and $r^i(h_t)$ describes the information reported to subject i at the end of round t. In round t, each subject chooses $h_t^i \in H^i$. At the end of round t, subject i will be told the information $r^i(h_t)$ about what happened. Then the next round will be played. A behavioral model must explain how the sequence of choices for i, $(h_1^i, h_2^i, \ldots, h_R^i)$ is made, given what i knows at each round t[10].

The primary variables of our behavioral model are a finite set of remembered strategies for each agent i at each round t, $\Theta_t^i \subset H^i$ and a probability measure, π_t^i on Θ_t^i. Θ_t^i consists of J alternatives[11]. In round t, each agent selects an alternative, $\theta_{j,t}^i$, randomly from Θ_t^i using the probability density π_t^i on Θ_t^i and then chooses the action $h_t^i = \theta_{j,t}^i$ One can think of (Θ_t^i, π_t^i) as inducing a mixed strategy on H^i at t. At the end of each round t, agents are told $r(h_t)$. At the beginning of the next round $t + 1$, each agent computes a new H_{t+1}^i and π_{t+1}^i. The updating is performed in the following way:

First, *experimentation* takes place. It introduces new alternatives and, this way, the new alternatives enter into the collection and that diversity is maintained. For each $j = 1, \ldots, J$, with probability[12] ρ, a new contribution is selected at random from H^i and replaces $\theta_{j,t}^i$. We use a normal density for this experimentation. For each J, the mean value of the distribution is set equal

[8] A number of applications in economics use the genetic algorithm (developed by Holland, 1970, 1974) to implement this idea. For example, see Arifovic (1996), Miller (1996), Marks (1998), Vriend (2000), Lux and Schonstein (2005), etc.

[9] Our approach follows most closely Arifovic (1994), but there have been a number of other individual learning applications, for example, Marimon *et al.* (1990), Vriend (2000), Lux and Hommes (2008).

[10] Since we use the identical algorithm that we used for our simulations with $N = 5$, our description closely follows the behavioral model presented in Arifovic and Ledyard (2011).

[11] J is a free parameter of IEL. In this paper we set $J = 200$.

[12] ρ is a free parameter of the behavioral model. In this paper we set $\rho = 0.033$, exactly the same number we have used in our other IEL papers.

https://doi.org/10.1017/S026988891200015X Published online by Cambridge University Press

to the value of the alternative, $\theta^i_{j,t}$ that is to be replaced by a new alternative. The standard deviation is set to 1.

Secondly, *imitation* takes place. It increases the frequencies of the alternatives that would have been good choices in previous rounds. It allows potentially better paying strategies to replace those that might pay less. How do we define the measure of 'potentially better paying strategies'? We let $u^i(\theta^i_{jt}|r^i(h_t))$ be the hypothetical utility of alternative J at t given the information $r^i(h_t)$. This measures the utility that i thinks she would have gotten had she played θ_j last time. In other words, $u^i(\theta_j|r^i_t)$ is entirely hypothetical and must be specified for each application. Given a hypothetical utility function, u^i, here is how imitation takes place. For $j = 1, \ldots, J$, $\theta^i_{j,t+1}$ is chosen as follows. Pick two members of Θ^i_t randomly (with uniform probability) with replacement. Let these be $\theta^i_{k,t}$ and $\theta^i_{l,t}$. Then

$$\theta^i_{j,t+1} = \left\{ \begin{matrix} \theta^i_{k,t} \\ \theta^i_{l,t} \end{matrix} \right\} \textit{if} \left\{ \begin{matrix} u^i(\theta^i_{k,t}|\mu^i_t) \geqslant u^i(\theta^i_{l,t}|r^i_t) \\ u^i(\theta^i_{k,t}|\mu^i_t) < u^i(\theta^i_{l,t}|r^i_t) \end{matrix} \right\}$$

Imitation for $t+1$ favors alternatives with a lot of copies at t and alternatives that would have paid well at t, had they been used. So it is a process with a form of averaging over past periods. If the actual contributions of others have provided a favorable situation for an alternative $\theta^i_{j,t}$ on average, then that alternative will tend to accumulate replicates in Θ^i_t (it is fondly remembered), and thus will be more likely to be actually used. Over time, the sets Θ^i_t become more homogeneous as most alternatives become copies of the best-performing alternative.

Third, *selection* occurs. Each contribution has the following probability of being selected[13]:

$$\pi^i_{k,t+1} = \frac{u^i(\theta^i_{k,t+1}|r^i_t) + \varepsilon^i_{t+1}}{\sum_{j=1}^{J} (u^i(\theta^i_{j,t+1}|r^i_t) + \varepsilon^i_{t+1})}$$

for all $i \in \{1, \ldots, N\}$ and $k \in \{1, \ldots, J\}$ and where[14]

$$\varepsilon^i_{j,t+1} = \min_{\theta \in \Theta^i_{t+1}} \{0, u^i(\theta|r^i_t)\}$$

This completes the description of how IEL transitions from (Θ^i_t, π^i_t) to $(\Theta^i_{t+1}, \pi^i_{t+1})$, given the signal from the play in round t, r^i_t.

The only feature remaining to be specified is the *initialization* process—how (Θ^i_0, π^i_0) is determined. We implement two different approaches, *random* and *modified* initialization. With *random* initialization, for each i, we generate a set, Θ^i_0, of J messages using a uniform distribution on H^i. After that, we start the first period of the game. At $t = 1$, one of the alternatives that becomes an actual message is chosen randomly from the uniform distribution in $\{1, \ldots, J\}$.

We created *modified* initialization to try to capture the phenomenon that an agent, who thinks hard about the problem before beginning of the play, might be able to eliminate a lot of guessing and focus on productive strategies out of the box. We took an approach to this that is often referred to as level 1 of a cognitive hierarchy[15], where the individual assumes all others are behaving randomly while that individual optimizes against that randomness. In the modified initialization, the first stage is the same as with random initialization, that is, for each i, J messages are chosen from the uniform distribution over H^i. Second, for each j we draw, again randomly, 100 pairs of values of r_{-jk}. The payoff of message j is then calculated as $\upsilon_j = (1/100) \sum_k u^i(\theta^i_j|r_{-jk})$. It is the expected payoff of θ^i_j given r_{-jk}.

[13] An alternative selection model is the probabilistic choice function $\pi(\theta^k) = e^{\lambda u^i(\theta^k)} / \sum_j e^{\lambda u^i(\theta^k)}$. We have found (see e.g. Arifovic & Ledyard, 2011) that the behavior predicted changes very little with this model from our proportional selection rule, for all λ. This is because the set A tends to become homogeneous fairly fast, at which point the selection rule is irrelevant.

[14] This implies that if there are negative forgone utilities in a set, payoffs are normalized by adding a constant to each payoff that is, in absolute value, equal to the lowest payoff in the set.

[15] See Camerer and Chong (2004).

https://doi.org/10.1017/S026988891200015X Published online by Cambridge University Press

Third, imitation takes place based on these average payoffs. This gives us Θ_0^i. Finally, we select the actual message as above in selection.

At first glance one might suspect that our modified initialization is equivalent to just moving ahead one round from random initialization. But that is not quite right. It would be true if we just drew one pair of values of r_{-jk}. However, modified initialization essentially computes the best reply to the initial expected value of r_{-jk} instead of to a particular realization. This means that if the equilibrium value of r_{-jk} is near its initial expected value, then modified initialization will speed up convergence. However, if the equilibrium value of r_{-jk} is a ways away from its initial expected value, modified initialization may delay convergence. We expand on this idea more in Section 3.3 under 'The impact of the modified initialization'.

As reported in Arifovic and Ledyard (2011), with $N = 5$, IEL with modified initialization is somewhat faster in terms of convergence times. IEL with random initialization matched the data from Chan and Tang's experiments better, while the modified initialization matched the data from our own experiments better[16].

3 Procedures and results

3.1 The basics

The environment. In Arifovic and Ledyard (2011) we used the utility functions and cost of production in Chen and Tang (1998) in order to compare our results to those generated with humans. The per-person cost, c, of producing a unit of the public good, which determines Z, is set to 20 and the utility parameters are given in Table 1. We consider a basic environment E with $N = 5$. We also consider $E(10)$ with $N = 50$ and $E(20)$ with $N = 100$.

The behavioral model. We used the IEL model described in the previous section. For the runs reported in this section, we set the memory capacity $J = 200$ and $\Theta^i = [-4,6]$[17]. We set the rate of experimentation $\rho = 0.033$. Forgone utility $u^i = W^i(m^i|r^i(m), \gamma)$ from Equation (1). We consider both random and modified initialization.

The mechanisms. Our goal is to investigate the pattern of convergence times over a range of values of γ that includes the cut points associated with the dominant-diagonal and strategic-complementarity conditions. For our model with $N = 5$, the dominant-diagonal condition is satisfied for $\gamma \geqslant 30$, and the strategic-complementarity condition is satisfied for $\gamma \geqslant 80$. Thus, for $N = 5$, we picked $\gamma = 10$, 30, 50, 80, 100, 180, and 260.

The sufficient condition generated by a strategic-complements sufficiency condition for convergence of best-reply algorithms requires that γ be scaled linearly in N. This is also true for the dominant-diagonal condition for convergence of best-reply algorithms. Thus, for $N = 50$ ($k = 10$) and $N = 100$ ($k = 20$) we chose values of γ so that the ratio γ/k remained constant[18].

Table 1 Utility parameters

Agent	1	2	3	4	5
A_i	26	104	38	82	60
B_i	1	8	2	6	4
α_i	200	10	160	40	100

[16] Our subjects were given a 'what-if' calculator that they could use prior and during the beginning of an experiment. Chen and Tang's subjects did not have access to such a tool.

[17] This was the range of values used in the both the Arifovic–Ledyard and Chen-Tang experiments.

[18] It is worth pointing out that we use our learning algorithm to 'locate' equilibria. This is different from for solving for an evolutionary stable equilibrium, where one takes an existing equilibrium and asks whether it is stable with respect to best-response dynamics. We thank our anonymous referee for this remark.

https://doi.org/10.1017/S026988891200015X Published online by Cambridge University Press

The simulations. For each mechanism, γ, we conducted $R = 1000$ runs. Each run, r, was terminated 100 periods after a *convergence criterion* was fulfilled[19]. For the analysis of the behavior observed in the simulations reported in this paper, we adopt the same convergence criteria that we used in Arifovic and Ledyard (2011). This allows us to compare the results of simulations with 'small' and 'large' N.

Our convergence criterion is defined in terms of how close all agents' messages are to the equilibrium messages. This convergence criterion is fulfilled when the difference between the equilibrium value and the value of the selected message of each agent is less than or equal, in absolute terms, to a positive number d; that is, when $|m_t^i - m^{ie}| \leqslant 0.1$ for all i. In our simulations, we set $d = 0.1$.

3.2 Performance measures

Because we are interested in whether and how fast these mechanisms converge to the equilibrium, we use two measures of performance: (1) the time of first passage through equilibrium and (2) an index of equilibrium stability.

Time of convergence. In Arifovic and Ledyard (2011) the time of convergence is defined in the following way. The period when the convergence criterion is first fulfilled is called *the time of the first passage through equilibrium*, $T^{\gamma,r}$ for run r and given γ. The average time of the first passage through equilibrium for R runs, $\bar{T}^{\gamma}$, is given by:

$$\bar{T}_c^{\gamma} = \frac{\sum_{r=1}^{R} T^{\gamma,r}}{R}$$

We denote the standard deviation from this value, across the R runs, by $\sigma_{T^{\gamma}}$.

Stability of convergence. In addition to recording the time when our convergence criterion is first fulfilled, we also want to find out how stable it is (as in Arifovic & Ledyard, 2011). In other words, is the convergence criterion satisfied by accident and agents have heterogenous collections of messages, or do the collections converge towards the values of the equilibrium messages? The measure that we use (see Arifovic & Ledyard, 2011) is based on the individual's action sets—the set of strategies from which the agents choose their messages. We call this the *index of equilibrium stability in strategies* S^s. It measures the percentage of all possible choices that are close to that agent's equilibrium message.

$$S^s = \frac{\sum_{t=T^{\gamma,r}+1}^{T^{\gamma,r}+100} \sum_{i=1}^{N} \sum_{j=1}^{J} S_{j,t}^i}{NJ}$$

where $S_{j,t}^i$ is an index that equals 1 if $|a_{j,t}^i - m^{ie}| \leqslant d$ and otherwise equals 0.

3.3 Results

Convergence times. Tables 2–7 and Figure 1 show that increases in the number of agents do not affect the dynamics of IEL in GL games. The U-shaped feature of convergence time is preserved across 10- and 20-times increases in the number of agents. The factor $\gamma/N = 50$ remains the value that results in the minimum time to convergence.

Overall, the times of convergence slightly increase with 10- and 20-fold increases in the number of agents. With random initialization, for each γ/k value, $\bar{T}_c^{\gamma}$ increases by three to four periods as we move from $N = 5$ (Table 2) to $N = 50$ (Table 4), and then another few periods when we go from $N = 50$ to $N = 100$ (Table 6). The factor $\gamma/N = 50$ remains the value that results in the minimum time to convergence for both $N = 50$ and $N = 100$. Modified initialization results in faster convergence for $N = 50$ (see Table 3). However, for $N = 50$ and $N = 100$ the times of convergence are

[19] The maximum number of periods for each run was set at $t_{max} = 1000$. If the convergence criterion is not fulfilled by that time, a run is terminated. All of our runs, for all γ's and N's converged within 1000 periods.

https://doi.org/10.1017/S026988891200015X Published online by Cambridge University Press

Table 2 Time to convergence, $N = 5$, $k = 1$: random initialization, $\rho = 0.033$

γ	γ/k	$\hat{T}_c^{\gamma}(\sigma_{T^{\gamma}})$	$S^s(\sigma_{S^s})$
10	10	26.10 (11.09)	98.23 (2.51)
30	30	11.47 (2.58)	99.14 (1.05)
50	50	10.63 (2.17)	99.18 (1.00)
80	80	11.73 (3.14)	99.11 (1.11)
100	100	14.75 (5.81)	99.03 (1.19)
180	180	39.97 (20.63)	98.77 (1.93)
260	260	72.76 (36.43)	98.78 (2.01)

Table 3 Time to convergence, $N = 5$, $k = 1$: modified initialization, $\rho = 0.033$

γ	γ/k	$\hat{T}_c^{\gamma}(\sigma_{T^{\gamma}})$	$S^s(\sigma_{S^s})$
10	10	23.61 (9.73)	98.30 (2.55)
30	30	10.64 (2.68)	99.10 (1.04)
50	50	9.89 (2.49)	99.15 (1.05)
80	80	10.77 (3.12)	99.15 (1.05)
100	100	12.63 (4.88)	99.06 (1.24)
180	180	29.34 (18.98)	98.91 (1.47)
260	260	54.80 (35.83)	98.78 (1.66)

Table 4 Time to convergence, $N = 50$, $k = 10$: random initialization, $\rho = 0.033$

γ	γ/k	$\hat{T}_c^{\gamma}(\sigma_{T^{\gamma}})$	$S^s(\sigma_{S^s})$
100	10	29.01 (4.60)	96.14 (2.09)
300	30	14.69 (2.98)	97.20 (1.82)
500	50	13.96 (2.80)	97.70 (1.49)
800	80	14.99 (3.01)	98.04 (1.43)
1000	100	18.30 (4.81)	98.06 (1.47)
1800	180	45.74 (15.18)	97.82 (1.74)
2600	260	76.00 (24.60)	97.66 (1.89)

Table 5 Modified initialization, $\rho = 0.033$: time to convergence, $N = 50$, $k = 10$

γ	γ/k	$\hat{T}_c^{\gamma}(\sigma_{T^{\gamma}})$	$S^s(\sigma_{S^s})$
100	10	28.71 (4.58)	96.02 (2.13)
300	30	16.17 (2.97)	97.23 (1.70)
500	50	15.44 (2.88)	97.66 (1.66)
800	80	15.99 (3.20)	97.97 (1.47)
1000	100	18.81 (4.68)	98.18 (1.43)
1800	180	43.92 (13.37)	97.65 (1.94)
2600	260	76.49 (19.78)	97.61 (2.03)

almost the same as for random initialization (see Tables 5 and 7). Figure 1 plots times to convergence for all of our values of N and for both random and modified initializations. The figure illustrates nicely how all of the U-shaped curves associated with different N's and initialization procedures are close to each other, reaching the minimum value at $\gamma/N = 50$.

https://doi.org/10.1017/S026988891200015X Published online by Cambridge University Press

Table 6 Time to convergence, $N = 100$, $k = 20$: random initialization, $\rho = 0.033$

γ	γ/k	$\hat{T}_c^{\gamma}(\sigma_{T^{\gamma}})$	$S^s(\sigma_{S^s})$
200	10	32.51 (3.98)	99.87 (0.01)
600	30	16.15 (3.03)	99.87 (0.01)
1000	50	15.37 (2.87)	99.87 (0.02)
1600	80	16.36 (3.07)	99.86 (0.02)
2000	100	19.90 (4.52)	99.85 (0.04)
3600	180	46.48 (10.94)	99.80 (0.06)
5200	260	78.58 (15.27)	99.79 (0.05)

Table 7 Time to convergence, $N = 100$, $k = 20$: modified initialization, $\rho = 0.033$

γ	γ/k	$\hat{T}_c^{\gamma}(\sigma_{T^{\gamma}})$	$S^s(\sigma_{S^s})$
200	10	32.71 (4.35)	99.87 (0.01)
600	30	17.96 (2.98)	99.87 (0.01)
1000	50	17.31 (2.93)	99.87 (0.02)
1600	80	17.89 (3.02)	99.87 (0.02)
2000	100	21.09 (4.31)	99.86 (0.03)
3600	180	45.45 (9.74)	99.80 (0.06)
5200	260	77.05 (13.32)	99.79 (0.06)

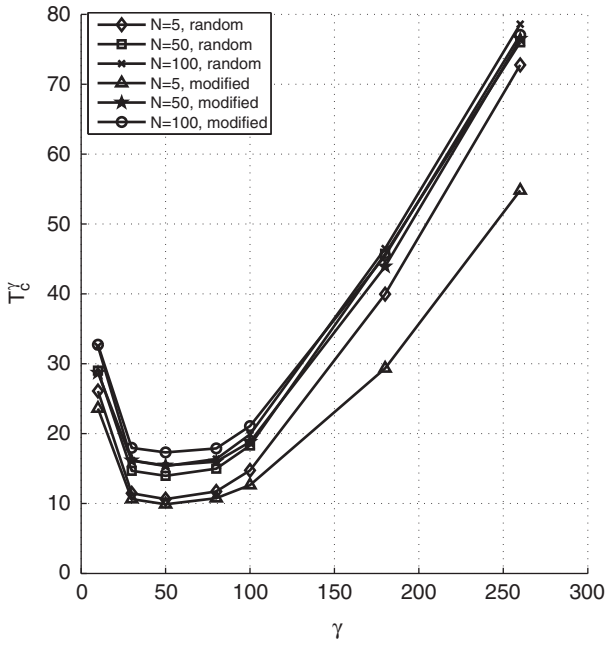


Figure 1 Convergence times for different N

It is perhaps understandable that for low values of γ/K, especially below 30, the time to converge will be higher than for values between 30 and 80 since best-reply dynamics are unstable (due primarily to overreactions) and so a period of averaging is necessary to get convergence. But this does not explain the fact that $\bar{T}_c^{\gamma}$ increases for high values of γ/k. As γ/k grows, there is stronger pressure for all agents to coordinate on a common value, typically not the equilibrium one, in order to minimize the difference between their own and average contribution of others.

https://doi.org/10.1017/S026988891200015X Published online by Cambridge University Press

Once the coordination takes place, strategies that participated in this receive relatively high payoffs and are then copied, increasing in frequency. So, collections of strategies become homogenized. At that point, experimentation is required in order to introduce strategies close to the equilibrium ones that will get the mechanism out of the non-equilibrium outcome. However, given the circumstances, it takes time for the required experimentation to succeed. Most new values that deviate from the strategy everyone has coordinated on will most likely receive lower forgone payoffs and, thus, disappear from the collections. The larger that γ/k is, the more important this effect becomes, increasing the time to converge to equilibrium.

The values of the standard deviation of the time to converge, σ_{T^γ}, follow the same U-shaped pattern as the times to convergence do. Standard deviations decrease with γ/k until $\gamma/k = 50$ (which is also the value of minimum $\bar{T}_c^\gamma$) and then start increasing, reaching high values for the two largest γ/k values, 180 and 260. This pattern is shared by both random and modified initialization. The standard deviations are smaller for the modified initialization than for the random initialization. The explanation for this is that all the simulations with modified initialization start out with collections of strategies that are more similar across different runs, and, thus, are likely to have more similar dynamics, and a tighter distribution of convergence times. The standard deviations decrease as N increases, a variation on the law of large numbers.

Stability. Once the IEL reaches an equilibrium, there is a high degree of stability. For all of the simulations, our measure of stability is above 96%. Again, one can notice, for $N = 5$ and $N = 50$ a slightly U-shaped pattern, with stability increasing as γ/k reaches 50, and slightly decreasing after that. Simulations with $N = 100$ have stability values close to 100% for all values of γ/k and for both types of initialization.

The invariance in scale. As long as we adjust $\gamma(k)$ so that $\gamma(k)/k$ is constant, the dynamics of IEL in GL mechanisms seem to be invariant in k. Why might this be so? We have two thoughts on this.

First, consider the standard theoretical convergence criteria based on the second-derivatives of the utility functions. Let $N = k\hat{N}, \gamma = k\hat{\gamma}$ and $\hat{\delta} = N/K = \hat{\gamma}/\hat{N}$. Strategic complements requires that $\gamma/N \geqslant 2B^i$ for all i. In the replica economy this is $\hat{\gamma}/\hat{N} \geqslant 2B^i$ for all i. Thus, once we adjust γ, the strategic-complements condition is scale independent. The dominant-diagonal condition is $\hat{\gamma}/\hat{N} \geqslant \frac{k\hat{N}-2}{k\hat{N}-1}B$. As k grows this is approximately $\hat{\gamma}/\hat{N} \geqslant B$, which is also scale independent.

Second, consider the utility payoff to any agent. It is symmetric around the best reply. Since IEL is picking messages that are on average in proportion to utility, then IEL is picking randomly around best reply. So on average IEL will be picking the best reply. Let us see what that means as $k \to \infty$. For quadratic utility functions, the best reply is $m = \frac{-(A-c)+[(\gamma/N)+2B](N-1)\mu}{(\gamma/N)[(N-1)-2NB]}$. This can be re-written as $m = \left[-\frac{1}{N-1}(A^i - c) + (\hat{\delta} + 2B)\mu\right] / \left[\hat{\delta}(1 - 2(\frac{k\hat{N}}{k\hat{N}-1})B)\right]$, which is approximately $m = -\phi + \psi\mu$, where $\phi = -(A-c)/[k\hat{N}\hat{\delta}(1-2B)]$ and $\psi = [\hat{\delta} + 2B]/[\hat{\delta}(1-2B)]$. As k grows, $k\mu_{t-1} = \frac{k}{k\hat{N}-1}(\hat{x}_{t-1} - m_{t-1}) \to x_{t-1}$. Therefore, the aggregate best response of the k individuals of type i is km_t and, as k grows, $km_t^i \to \frac{A-c}{(\hat{\delta}/\hat{N})(1-2B)} + \left[\frac{\hat{\delta}+2B}{\hat{\delta}(1-2B)}\right]x_{t-1} = \phi^i + \psi^i x_{t-1}$. Thus as k grows, the new $x_t = \left[\sum_{i=1}^{\hat{n}} \phi^i\right] + \left[\sum_{i=1}^{\hat{n}} \psi^i\right]x_{t-1}$ is scale independent as a function of x_{t-1}. This suggests that on average the rates of convergence are independent of scale as long as we keep $\hat{\delta}$ constant.

The impact of the modified initialization. Based on our work with $N = 5$, we expected the modified initialization to lead to faster times of convergence. But, instead, for $N = 50$ and $N = 100$ the convergence times are pretty much the same for random and modified initializations. There is an explanation for this.

As described in the previous section, we start the modified initialization by randomly drawing J rules for each agent i. Then for each rule j, we randomly draw 100 values of μ_{-jk} in $[-4,6]$ and 100 values of σ^2_{-jk} in $[0,5]$ and compute the average (expected) utility, $v_j^i = (1/100)\sum_k u^i(\theta_j^i | r_{-jk})$, for that rule. Then using these average utilities, we do imitation. For the games generated by GL mechanisms in quadratic environments, the parts of our hypothetical utility functions that are involved in comparing any two rules for imitation are linear in μ_{-i} and do not include σ^2_{-i}. Thus, the distribution of σ^2_{-i} does not matter at all, and only the expected value of $\mu_{-i} = 1$ plays a role.

https://doi.org/10.1017/S026988891200015X Published online by Cambridge University Press

So (with some randomness) our initialization essentially computes the best response to $\mu_{-i} = 1$. So the closer the expected value of μ_{-i} is to its equilibrium value, the faster convergence will be.

The precise calculation of the equilibrium value of μ_{-i} is

$$\mu_{-i} = \frac{\hat{X} - \hat{m}^i}{N-1} = \frac{1}{5k}\left\{\hat{X} - \frac{1}{\hat{\gamma}(5k-1)}\left[A^i - 2B^i\hat{X} - c\right]\right\}$$

For $N = 5$ and $\gamma = 10$, the equilibrium values of μ_{-i} are 1.1, 0.9, 1.05, 0.95, and 1. Fixing N, as γ/k grows, the equilibrium value of μ_{-i} goes to $\hat{X}/N$, which equals 1. So for all $\gamma \geqslant 10$, if $N = 5$, the equilibrium values of μ_{-i} hang around 1. Thus for $N = 5$, modified initialization starts everything out just right. But, with $\gamma = k\hat{\gamma}$, as k grows the equilibrium values of μ_{-i} go to zero. This means that the expected value of μ_{-i} drawn in the initialization is moving further away from its equilibrium value. For $k = 10$, $N = 50$, and $\gamma/k = 50$, the equilibrium values of μ_{-i} are near 0.1, and for $k = 20$ they are near 0.05. That is, they are approximately $1/k$ of the value for $N = 5$. So the advantage that modified initialization has for $N = 5$ is seriously eroded as k grows.

This intuition also helps explain why the variance of convergence times is lower for modified than for random. Modified starts out at about the same place each time, whereas random starts from many different places. On average they take the same time (since, on average, random also starts at $\mu_{-i} = 1$) but sometimes random is closer and sometimes it is not.

The impact of an increase in the rate of experimentation. We conducted a set of simulations with random initialization and a rate of experimentation for $\rho = 0.066$. The higher rate of experimentation, twice as high as in our baseline simulations, resulted in somewhat faster convergence times. On average, for each γ/k, convergence is by three to four periods faster compared with the simulations with $\rho = 0.033$. We illustrate the results, for $N = 5$ and random initialization in Table 7. It is interesting to note that relatively high rate of experimentation does not introduce disruption that might slow down or prevent convergence. Instead, it speeds things up. The reason for this is the fact that new values generated via experimentation are not played out right away. They first have to prove, in terms of forgone payoffs, that they might be worthwhile candidates. If they do so, then, through imitation, they increase in frequency and thus increase their chances to be selected as actual messages. Otherwise, new values with relatively low forgone payoffs quickly disappear from the collections (See Table 8 for times to convergence for $\rho = 0.066$).

Variation in the rate of experimentation. The behavior observed in case of a twice-as-high rate of experimentation as our baseline case, for $N = 5$, made us curious about the robustness of our large N system with respect to a wide range of rates of experimentation. Thus, in addition, we simulated the system for the following values of ρ: 0.0033, 0.0066, 0.01, 0.066, 0.1, and 0.2. Again, we used the same values of γ/k equal to 10, 30, 50, 80, 100, 180, and 260. We report the results (which are the averages over $R = 1000$ runs in Tables 9 (for $N = 5$), Table 10 (for $N = 50$), and Table 11 (for $N = 100$)). For brevity, we report the results for $\gamma/k = 10$, 50 and 260 only. Our results show that the U-shaped pattern of behavior is preserved regardless of the rate of experimentation. They also show that very low rates, such as 0.0033 and 0.0066, result in slower convergence. However, there

Table 8 Time to convergence, $N = 5$, $k = 1$: random initialization, $\rho = 0.066$

γ	γ/k	$\hat{T}^{\gamma}_{c}(\sigma_{T^{\gamma}})$	$S^{s}(\sigma_{S^{s}})$
10	10	23.61 (9.73)	98.30 (2.55)
30	30	10.64 (2.68)	99.10 (1.04)
50	50	9.89 (2.49)	99.15 (1.05)
80	80	10.77 (3.12)	99.15 (1.05)
100	100	12.63 (4.88)	99.06 (1.24)
180	180	29.34 (18.98)	98.91 (1.47)
260	260	54.80 (35.83)	98.78 (1.66)

https://doi.org/10.1017/S026988891200015X Published online by Cambridge University Press

Table 9 Different rates of experimentation, time to convergence, $N = 5$, $k = 1$

ρ	γ	γ/k	$\hat{T}_c^{\gamma}(\sigma_{T^{\gamma}})$	$S^s(\sigma_{S^s})$
$\rho = 0.0033$	10	10	66.67 (52.40)	99.40 (2.08)
	50	50	15.90 (19.29)	99.88 (0.42)
	260	260	223.48 (170.20)	99.65 (1.56)
$\rho = 0.0066$	10	10	44.26 (30.00)	99.55 (1.54)
	50	50	12.00 (10.25)	99.88 (0.27)
	260	260	132.97 (97.32)	99.75 (0.81)
$\rho = 0.01$	10	10	35.56 (21.44)	99.66 (0.84)
	50	50	10.51 (5.78)	99.88 (0.18)
	260	260	103.39 (70.82)	99.78 (0.55)
$\rho = 0.066$	10	10	19.55 (7.31)	99.26 (0.40)
	50	50	8.79 (1.80)	99.45 (0.13)
	260	260	45.98 (27.49)	99.35 (0.27)
$\rho = 0.1$	10	10	18.70 (6.20)	98.51 (0.43)
	50	50	8.78 (1.59)	98.79 (0.16)
	260	260	41.55 (24.26)	98.66 (0.37)
$\rho = 0.2$	10	10	19.57 (6.25)	92.73 (0.77)
	50	50	9.87 (1.89)	94.44 (0.25)
	260	260	39.75 (23.18)	94.01 (0.47)

Table 10 Different rates of experimentation, time to convergence, $N = 5$, $k = 10$

$\rho = 0.0033$	γ	γ/k	$\hat{T}_c^{\gamma}(\sigma_{T^{\gamma}})$	$S^s(\sigma_{S^s})$
$\rho = 0.0033$	100	10	99.35 (32.35)	99.97 (0.11)
	500	50	63.09 (31.83)	99.98 (0.03)
	2600	260	508.38 (146.58)	99.98 (0.03)
$\rho = 0.0066$	100	10	59.55 (17.49)	99.97 (0.05)
	500	50	36.14 (15.56)	99.98 (0.02)
	2600	260	290.27 (81.92)	99.97 (0.05)
$\rho = 0.01$	100	10	45.27 (12.18)	99.97 (0.02)
	500	50	26.76 (9.95)	99.97 (0.02)
	2600	260	214.57 (62.09)	99.97 (0.03)
$\rho = 0.066$	100	10	23.56 (3.43)	99.52 (0.02)
	500	50	12.38 (1.65)	99.54 (0.01)
	2600	260	80.11 (21.87)	99.53 (0.02)
$\rho = 0.1$	100	10	23.89 (3.62)	98.84 (0.03)
	1000	50	12.21 (1.43)	98.91 (0.02)
	2600	260	73.99 (19.61)	98.90 (0.02)
$\rho = 0.2$	100	10	46.81 (18.38)	93.71 (0.23)
	500	50	19.33 (7.12)	94.80 (0.04)
	2600	260	78.42 (21.54)	94.78 (0.05)

is a difference between the two. The rate of $\rho = 0.0033$ results in the highest average times of convergence. These averages decrease almost in half for $\rho = 0.0066$.

For $N = 5$, higher rates of experimentation of 0.1 and 0.2 do not have much of an impact on the times to convergence compared with our baseline case. In addition, for $N = 50$ and $N = 100$, the higher rates of experimentation of $\rho = 0.066$ and 0.1 do not have much impact on the average times to convergence compared with the baseline case with $\rho = 0.033$. However, a high rate of experimentation, $\rho = 0.2$, results in somewhat higher values of the average times to convergence. The effects of this high rate of experimentation are stronger in case of $N = 100$. Note that the qualitative features of the

https://doi.org/10.1017/S026988891200015X Published online by Cambridge University Press

Table 11 Different rates of experimentation, time to convergence, $N = 100$, $k = 20$

ρ	γ	γ/k	$\hat{T}_c^{\gamma}(\sigma_{T^{\gamma}})$	$S^s(\sigma_{S^s})$
$\rho = 0.0033$	200	10	120.51 (33.00)	99.99 (0.01)
	1000	50	83.47 (31.82)	99.99 (0.01)
	5200	260	575.36 (103.26)	99.99 (0.03)
$\rho = 0.0066$	200	10	69.13 (15.92)	99.98 (0.01)
	1000	50	45.95 (15.04)	99.98 (0.01)
	5200	260	323.53 (58.70)	99.98 (0.01)
$\rho = 0.01$	200	10	51.75 (11.19)	99.98 (0.01)
	1000	50	33.10 (9.78)	99.98 (0.01)
	5200	260	236.39 (41.80)	99.98 (0.02)
$\rho = 0.066$	200	10	26.19 (2.80)	99.54 (0.01)
	1000	50	13.81 (1.71)	99.54 (0.01)
	5200	260	87.47 (14.70	99.54 (0.01)
$\rho = 0.1$	200	10	27.24 (3.37)	98.90 (0.01)
	1000	50	13.56 (1.71)	98.91 (0.01)
	5200	260	79.66 (13.51)	98.91 (0.01)
$\rho = 0.2$	200	10	159.65 (128.34)	94.41 (0.10)
	1000	50	63.03 (1.39)	94.80 (0.03)
	5200	260	122.80 (1.39)	94.79 (0.03)

behavior are still preserved even when the collection of strategies undergo this fairly large experimentation, with 20% turnover in each time period. Finally, it is worthwhile to note that a relatively high rate of experimentation of 0.2 still does not affect the stability once the convergence occurs. For all N the stability goes down by few percentage points, but remains well above 90%.

3.4 Comparison with other models of learning

A number of models of individual learning have been developed over the past decade. (For an excellent overview, see Camerer, 2003). Much of the research has been done in the context of one-shot games with small strategy spaces such as 2-by-2 or 3-by-3 games. The performance of the models has generally been evaluated by using standard econometric methods (maximum likelihood or grid search) to fit the models to experimental data. Two of the most frequently used models from this strand of literature are the Reinforcement Learning (RL) (Erev & Roth, 1998) and the Experience-Weighted Attraction (EWA) Learning (Camerer & Ho, 1999).

The implementation of both algorithms requires either that all possible players' strategies are enumerated, that is, explicitly represented in the collection, or that the strategy space is discretized. In RL, strategies that achieve higher returns when used are *reinforced* and played with a higher probability. In EWA, each strategy has an *attraction* based on the possible payoff it might have earned had it been played in the past. Strategies with higher attractions have higher probabilities of being selected. One of the primary differences between RL and EWA is the latter's use of hypothetical computations to quickly evaluate all strategies. RL only uses actual payoffs and thus can only evaluate strategies that have actually been played.

What is common to RL and EWA on one hand, and IEL on the other, is that all of them update the collections of their strategies in such a way that the frequencies of those that have performed well increase over time. The choice of a particular strategy as the actual strategy that a player uses in a given period is probabilistic, and the strategies' selection probabilities depend positively on their past performance.

The differences between IEL, on one hand, and RL and EWA on the other, stem from the way that the strategy sets (collections) are determined and updated. IEL starts out with a set of randomly generated strategies (messages). It brings in new strategies to be tried via experimentation. The way

https://doi.org/10.1017/S026988891200015X Published online by Cambridge University Press

experimentation is implemented allows IEL, unlike RL and EWA, to handle large strategy spaces well. For example, in the GL mechanism, each agent has a continuum of possible messages. In order to apply models such as RL and EWA, the continuum must be discretized. However, discretization causes problems when there are very fine differences in equilibrium values between different mechanisms. IEL handles that problem well. It does start out with randomly chosen sets of alternatives for each agent, but due to directed experimentation there is a sufficiently high probability that any important omitted messages, such as the Nash Equilibrium messages, will be added to the set.

IEL adds a new dimension to RL and EWA by allowing agents to vary their active strategy set in response to experience. What has been 'learned' by an agent at any time is summarized not in attraction weights but in the set of active strategies. Strategies that have been or would have been successful will have more copies in the active strategy set. If a strategy has a lot of copies in the active set, it will be chosen with a higher probability. The primary difference between IEL and RL and EWA seems to be that IEL discards strategies that aren't potentially profitable and thus does not waste time or lose payoffs re-testing unprofitable options. (See Arifovic and Ledyard (2004) for the comparison of the performance of RL, EWA, and IEL in the GL mechanisms. Also, see Arifovic and Ledyard (2009) for comparison of the performance of IEL and EWA in a voluntary contribution mechanism.)

4 Final remarks

In Arifovic and Ledyard (2011), we investigated IEL's behavior in a class of games with a *small* number of agents. The number of agents in that setup was equal to five in order to match the design and the number of human subjects who participated in the laboratory experiments. Both IEL simulations and experiments were conducted for a number of different values of the free parameter of the model. Our results showed that IEL successfully captures and predicts the behavior observed in the experiments with human subjects. We have also successfully applied our behavioral model in the context of the call markets where the IEL model generates the same types of price volatility and efficiencies as those generated in our experiments with human subjects (Arifovic & Ledyard, 2011), and to the voluntary provision of public goods environment where our model matches the patterns of behavior of the average contribution over time from a number of different human subjects experiments (Arifovic & Ledyard, 2011). This collection of results suggests that we can use our behavioral model as a computer testbed to study a number of mechanism design issues such as out-of-equilibrium behavior, speed of convergence, efficiency, price volatility, and the stability of equilibria.

Thus, in this paper, we used this methodology to investigate what happens with GL mechanisms in linear public-good environments when the number of agents, N, becomes large. Testbeds of this scale are not easily implemented in the controlled laboratory setting with human subjects. Our findings are interesting. The main features of the behavior observed with the small number of agents are preserved when γ is scaled by the multiplicative factor k. For a given γ/k, the average times to convergence are similar. Further, the U-shaped pattern arising from variations in the value of γ/k is preserved. This result is robust to implementation of a wide range, low and high, of rates of experimentation. The advantage of IEL over other models of individual learning, such as RL and EWA Learning is that it can handle large strategy spaces well.

Acknowledgments

We thank Olena Kostyshyna for her very able research assistance. We also thank an anonymous referee for very helpful comments.

References

Arifovic, J. 1994. Genetic algorithm learning and the cobweb model. *Journal of Economic Dynamics and Control* **18**, 3–28.

Arifovic, J. 1996. The behavior of the exchange rate in the genetic algorithm and experimental economies. *Journal of Political Economy* **104**, 510–541.

https://doi.org/10.1017/S026988891200015X Published online by Cambridge University Press

Arifovic, J. & Ledyard, J. 2004. Scaling up learning models in public good games. *Journal of Public Economic Theory* **6**, 205–238.
Arifovic, J. & Ledyard, J. 2009. Individual Evolutioanry Learning and the Voluntary Contributions Mechanisms. manuscript.
Arifovic, J. & Ledyard, J. 2011. A behavioral model for mechanism design: Individual Evolutionary Learning. *Journal of Economic Behavior and Organization* **78**, 374–395.
Camerer, C. & Ho, T. 1999. Experience weighted attraction learning in normal form games. *Econometrica* **67**, 827–873.
Camerer, C. F. 2003. *Behavioral Game Theory: Experiments in Strategic Interaction*. Princeton University Press.
Camerer, C. & Chong, J. 2004. A cognitive hierarchy model of games. *The Quarterly Journal of Economics* **119**, 861–898.
Chen, Y. & Plott, C. 1996. The Groves–Ledyard mechanism: an experimental study of institutional design. *Journal of Public Economics* **59**, 335–364.
Chen, Y. & Tang, F. 1998. Learning and incentive compatible mechanisms for public goods provision: an experimental study. *Journal of Political Economy* **106**, 633–662.
Erev, I. & Roth, A. 1998. Predicting how people play games: reinforcement learning in experimental games with unique, mixed strategy equilibria. *American Economic Review* **80**, 848–881.
Gabay, D. & Moulin, H. 1980. On the uniquencess and stability of Nash equilibria in noncooperative games. In *Appliced Stochastic Control in Econometrics and Management Science*, Bensoussan, A., Kleindorfer, P.S., & Tapiero, C.S. (eds), North-Holland, Amsterdam, pp. 271–293.
Groves, T. & Ledyard, J. 1977. Optimal allocation of public goods: a solution to the 'Free Rider' problem. *Econometrica* **45**, 783–809.
Healy, P. 2006. Learning dynamics for mechanism design: an experimental comparison of public goods mechanisms. *Journal of Economics Theory* **129**, 114–149.
Holland, J.H. 1970. Robust algorithms for adaptation set in a general formal frameowork. *Proceedings of the IEEE Symposium on Adaptive Processes, Decision and Control* **17**, 51–55.
Holland, J. H. 1974. *Adaptation in Natural and Artificial Systems*. University of Michigan Press.
Hommes, C. & Lux, T. 2008. *Individual Learning, Heterogeneity and Aggregate Behavior in Cobweb Experiments*. Manuscript.
Lux, T. & Schornstein, S. 2005. Genetic learning as an explanation of stylized facts of foreign exchange markets. *International Journal of Mathematical Economics* **41**, 169–196.
Marimon, R., McGrattan, E. & Sargent, T.J. 1990. Money as a medium of exchange in an economy with artificially intelligent agents. *Journal of Economic Dynamics and Control* **14**, 329–373.
Marks, R. E. 1998. Evolved perception and behavior in oligopolies. *Journal of Economic Dynamics and Control* **22**, 1209–1233.
Milgrom, P. & Roberts, J. 1990. Rationalizability, learning, and equilibrium in games with strategic complementarities. *Econometrica* **58**, 1255–1277.
Miller, J. 1996. The coevolution of automata in the repeated prisonner's dilemma. *Journal of Economic Behavior and Organization* **29**, 87–112.
Muench, T. & Walker, M. 1983. Are Groves–Ledyard equilibria attainable? *The Review of Economic Studies* **50**, 393–396.
Page, S. & Tassier, T. 2004. Equilibrium selection and stability for the Groves–Ledyard mechanism. *Journal of Public Economic Theory* **6**, 311–335.
Vriend, N. 2000. An illustration of the essential difference between individual and social learning and its consequences for computational analyses. *Journal of Economic Dynamics and Control* **24**, 1–19.
Watkins, C. 1989. *Learning from Delayed Rewards*. University of Cambridge.

https://doi.org/10.1017/S026988891200015X Published online by Cambridge University Press