Evolutionary Dynamics

Introduction

Man's unprecedented rise to evolutionary dominance is rooted in his ability to variously cooperate with, punish, or take advantage of his fellow agents. One way to explore these life-governing dynamics is to consider a system of agents that evolves over time, where each generation of agents adopts properties of the strongest or most successful agents of the previous generation. In this paper, we consider two models of this idea: replicator dynamics, where agents are randomly paired with other agents in the population, and imitator dynamics, where agents are matched with proximate agents. We test each model with three well-studied, 2x2 normal-form games: the Prisoner's Dilemma, the Stag Hunt, and the Battle of the Sexes. We analyze how different initializations, dynamics, and hyperparameters affect an evolutionary multi-agent system.

Replicator Dynamics

We first considered a collection of 900 agents that are randomly paired to play each game. We consider the outcome of a game as the sum of the payoffs received if that game were infinitely repeated, given some discount rate (γ) proportionally diminishing the payoff for each successive round. After 5,000 games have been played in this fashion, new batches of agents are created for each agent type. Per the replicator dynamic, the change in the relative size of each agent group is determined by the group's current proportion of the population, multiplied by the difference between the average utility of that group and the average utility of all agents. We refer to one cycle of game play and agent evolution as a generation.

We considered four agent strategies, or agent types, including: *Always Cooperate* (AC), *Always Defect* (AD), *Tit-for-tat* (TFT), and *Not-Tit-for-Tat* (nTFT). We tested all possible two, three, and four agent combinations, while varying initial proportions for each test. Each of these variants was examined for each of the three games using discount rates of 0.95 and 0.99.

Because our results depend on random pairings, we report the average values from five repetitions of each experiment. Each experiment was run for a maximum of 30 generations, or until only one type of agent remained.[1] We variously refer to this as that agent surviving or dominating. If the agent initially constituted a small proportion of the system or nation, we say that that agent has invaded.

---

[1] This is a sufficient condition for what we will refer to as convergence, equilibrium, or steady-state.

AD vs TFT in Prisoner's Dilemma

When the population comprises AD and TFT for the Prisoner's Dilemma, the initial relative proportions effectively determines how many generations AD survives. Figure 1 shows that TFT is able to eradicate AD in most cases. Notably, when there is a much larger percentage of AD players (90%), it takes considerably longer for the TFT community to gain a foothold in the AD nation.

As expected, we found that AD did better with a smaller discount rate. This is because when TFT and AD play, AD receives a higher payoff than TFT only for the first round. With a smaller discount rate, the payoff for the first game is weighted relatively more, giving AD a bigger advantage in the head-to-head matchup.

The results in Figure 1 were obtained with γ = .95. When γ = .99, AD lost ground to TFT much more quickly, even when it started with a large majority. This shows that the ability for TFT to invade or dominate strongly depends on our choice of γ; that is, provided we have two or more TFT agents playing each other each round, a sufficiently large γ will support a TFT invasion of an AD nation.[2]

Figure 1: Ratios of AD vs TFT and final state of system, γ=.95

Initialization	Final state
10% AD, 90% TFT	0% AD, 100% TFT
25% AD, 75% TFT	0% AD, 100% TFT
50% AD, 50% TFT	0.1% AD, 99.9% TFT
75% AD, 25% TFT	1.4% AD, 98.6% TFT
90% AD, 10% TFT	77% AD, 23% TFT

Including other agent types in Prisoner's Dilemma

When we include AC and nTFT in Prisoner's Dilemma, we found that AD was able to exploit their comparatively more cooperative strategies. This enabled AD to feed off of them and survive in larger numbers.

Figure 2 demonstrates the effect of adding AC and nTFT. When the initial population consists of equal proportions of AD, TFT, and AC/nTFT, AD is able to survive after 30 generations, where originally it could not (ending 5.7% in the equal population test). This is because AD is able to consistently exploit AC and nTFT, capturing the highest

[2] A more granular analysis of systems with the initial AD population greater than 90% might reveal its associated basin of attraction.

payoff for several generations. However, as this “feeder” population depletes, AD is once more slowly eroded by TFT. Smaller feeder populations (e.g. 10% AC) left the AD group almost totally depleted, not unlike when AD faced TFT alone. Nevertheless, the larger the initial population of these feeders types, the larger share AD can capture. Varying γ from .95 to .99 had less of an effect here than in the previous experiment. The larger γ value left TFT with a slightly larger population on average.

Figure 2: Ratios of AD and TFT when combined with other agents

Initialization	Final State
25% AC, 50% AD, 25% TFT	.3% AC, 9.9% AD, 89.8% TFT
25% AC, 25% AD, 50% TFT	9.4% AC, 2.4% AD, 88.2% TFT
33.3% AC, 33.3% AD, 33.3% TFT	1.9% AC, 5.7% AD, 92.4% TFT
20% AC, 40% AD, 40% TFT	4.1% AC, 1.5% AD, 94.4% TFT
10% AC, 45% AD, 45% TFT	4% AC, .4% AD, 95.6% TFT
50% AD, 25% TFT, 25% nTFT	9.1% AD, 90.9% TFT, 1% nTFT
25% AD, 50% TFT, 25% nTFT	.6% AD, 96.9% TFT, 2.5% nTFT
33.3% AD, 33.3% TFT, 33.3% nTFT	3.1% AD, 96% TFT, .9% nTFT
40% AD, 40% TFT, 20% nTFT	.8% AD, 97.6% TFT, 1.6% nTFT
45% AD, 45% TFT, 10% nTFT	.2% AD, 98.9% TFT, .9% nTFT

Stag Hunt

The Stag Hunt possesses dynamics similar to the Prisoner’s Dilemma. The main difference is that the highest possible payoff for the Stag Hunt requires cooperation from both parties, while the reward for defecting is smaller and independent of the other agent’s action. Consequently, we would expect that cooperative agents are more likely to succeed.

Our results demonstrated this. Figure 3 shows that AD cannot succeed against TFT for most initializations. When AC is also included, we found that TFT and AC can both thrive, while AD is eroded.

Figure 3: AD's inability to compete using exploitation

Initialization	Final state
10% AD, 90% TFT	0% AD, 100% TFT
25% AD, 75% TFT	0% AD, 100% TFT
50% AD, 50% TFT	0% AD, 100% TFT
75% AD, 25% TFT	0% AD, 100% TFT
90% AD, 10% TFT	.5% AD, 99.5% TFT

Battle of the Sexes

This game had to be formulated differently, as it does not share the payoff symmetry on the diagonal that the previous games exhibited. Put another way, players receive different payoffs if they choose the same action, and the same payoffs if they choose different actions.

Because "defect" and "cooperate" do not manifest themselves the same way as they do in the Prisoner's Dilemma, we found it necessary to clarify the definitions of the agents for this game,:

- *Always cooperate* always chooses the other player's preferred action.
- *Always defect* always chooses its own preferred action.
- *Tit-for-tat* always begins by choosing the other player's preferred action, and then chooses whichever action their opponent picked for the previous round.[3]
- *Not tit-for-tat* begins by choosing its preferred action, then repeats doing the opposite of whatever its opponent did on the previous round.

Using these definitions, these agents cannot cooperate in the same way they can in the other games, and, in particular, no agent can effectively cooperate with itself. Consequently, we found that AC performs poorly. In the other games, AC agents can cooperate together and obtain a high cooperative payoff. In Battle of the Sexes, we have defined cooperating to mean that each agent chooses its opponent's preferred activity. Thus, the agents are rewarded with the lowest payoff for their altruism.

[3] That is, they mimic their opponent's action (e.g. choice of venue), and not whether it "cooperated" or "defected". We might alternatively define TFT to mimic its opponent's choice relative to TFT's opponent's preference. That is, if TFT is paired with AC, both choose their opponent's preferred venue, always receiving a payoff of 0.

Again because of our definitions, we found that AD is more successful than TFT. Whereas previously, TFT punishes an agent for defecting, here it acquiesces to its opponent's action, giving it a lower, but non-zero payoff. In effect, AD can "bully" TFT to cooperate with AD's preferred outcome, giving an AD agent the higher payoff for any particular head-to-head game against a TFT agent.

Notably, neither agent can crowd out the other, since they depend on each other for survival. Indeed, we have discovered our first "symbiotic" relationship, where agents cannot succeed alone, but must team with other "compatible" agents to survive. Figure 4 shows that systems with only AD and TFT converge to about 60% AD and 40% TFT, regardless of initialization. This is notable because it represents the ratio of AD's preferred payoff (3) to TFT's less preferred payoff (2).

Figure 4: AD and TFT equilibrium

Initialization	Final state
10% AD, 90% TFT	59.8% AD, 40.2% TFT
25% AD, 75% TFT	60% AD, 40% TFT
50% AD, 50% TFT	59.7% AD, 40.3% TFT
75% AD, 25% TFT	60.1% AD, 39.9% TFT
90% AD, 10% TFT	59.7% AD, 40.3% TFT

Figure 5 shows the AD population consistently outperforms the other agents as well, as it bullies AC and TFT, though stalemates against nTFT. However, because nTFT cannot cooperate with any agent, it was quickly pushed out of the population. We then observe that AD consistently converged to 60% of the population space, the same as in the previous experiment, with the leftover space distributed between AC and TFT.[4]

[4] In these experiments, we found that altering γ had no material effect.

Figure 5: No one agent dominates in the Battle of the Sexes, γ = .95

Initialization	Final State
25% AC, 25% AD, 25% TFT, 25% nTFT	13.6% AC, 59.9% AD, 26.8% TFT
25% AC, 50% AD, 25% TFT	18.5% AC, 60.2% AD, 21.3% TFT
25% AC, 25% AD, 50% TFT	13.4% AC, 59.9% AD, 26.7% TFT
40% AC, 20% AD, 20% TFT	19.6% AC, 59.7% AD, 20.7% TFT
33.3% AC, 33.3% AD, 33.3% TFT	18% AC, 60.1% AD, 21.9% TFT
20% AC, 40% AD, 40% TFT	14.2% AC, 59.6% AD, 26.2% TFT
10% AC, 45% AD, 45% TFT	9.1% AC, 59.9% AD, 31% TFT
50% AD, 25% TFT, 25% nTFT	59.9% AD, 40.1% TFT, 0% nTFT
25% AD, 50% TFT, 25% nTFT	60.1% AD, 39.9% TFT, 0% nTFT
40% AD, 20% TFT, 20% nTFT	60.1% AD, 39.9% TFT, 0% nTFT
33.3% AD, 33.3% TFT, 33.3% nTFT	59.90% AD, 40.1% TFT, 0% nTFT
40% AD, 40% TFT, 20% nTFT	60.3% AD, 39.7% TFT, 0% nTFT
45% AD, 45% TFT, 10% nTFT	59.8% AD, 40.2% TFT, 0% nTFT

Imitator Dynamics

We next considered agents on a 30x30 lattice. In lieu of random pairings, we consider the net payoff an agent receives for playing each of its 8 neighbors in the infinitely repeated game.[5] For each generation, each agent adopts the strategy of the neighboring agent with the highest payoff if that payoff is strictly greater than its own. If multiple neighbors receive the maximal payoff, the central agent randomly chooses one to imitate.

We report both which agents survive when convergence is reached and the number of rounds it took, while varying the game-type, lattice initialization, and discount factor. Figure 7 summarizes these results for the Prisoner's Dilemma and the Stag Hunt, where convergence was generally reached.

[5] Agents on the edges are paired with corresponding agents on the opposite edges, effectively obviating our "edge" cases.

Figure 7: Survivors given convergence on 200 iterations[6]

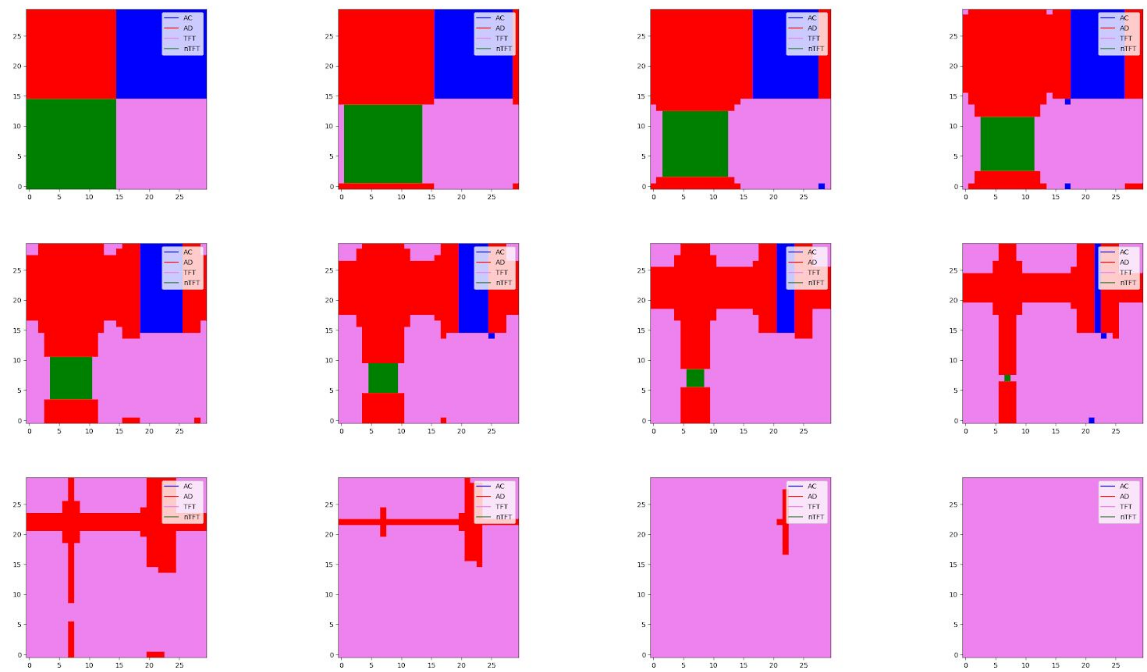
Game	Survivors	Observations	% of observations	Avg. generations before convergence	Std. dev.	Average initialization			
						% AC	% AD	% TFT	% nTFT
Prisoner's Dilemma	AC, TFT	7	4%	9.6	3.5	23%	17%	55%	5%
	AD	56	28%	3.1	1.4	31%	28%	12%	29%
	TFT	135	68%	12.4	3.8	22%	27%	29%	22%
	Divergent	2	1%	-	-	46%	7%	44%	4%
Stag Hunt	AC, TFT	179	90%	2.8	1.9	28%	23%	26%	24%
	TFT	19	10%	4.3	1.8	5%	38%	22%	35%
	Divergent	2	1%	-	-	9%	26%	2%	64%

From this, we observe that TFT is generally the most successful agent. In cases where TFT has an unfavorable initialization, AD can survive in the Prisoner's Dilemma (28% of tests), but not in the Stag Hunt. In rare instances, TFT and AC survive in the Prisoner's Dilemma (4%), whereas they both usually survive in the Stag Hunt (90%).

One unique feature of the lattice approach is the outcome depends not only on the initialization ratio, but also the placement. Consequently, we considered several initialization schemes where the agents were initialized in "communities", i.e. adjacent to their peers. One approach is to divide the lattice into quadrants. Figure 8 shows how TFT eventually dominates this initialization, but only after AD aggressively consumes AC. In each of the following figures, red represents **AD**, blue represents **AC**, green represents **nTFT**, and pink represents **TFT**.

[6] γ of.95 and .99 were each used for 100 iterations. Results have been consolidated here because It did not appear to have a material effect. To the extent there was an effect, γ=.99 appeared to more strongly favor the most probable survivor/group as we might expect.

Figure 8: TFT wins the Quadrant Prisoner's Dilemma



Interestingly, TFT's willingness to cooperate with AC can be its undoing, as it will not convert the AC agents (provided both are generally cooperating with their neighbors). This feature of TFT may allow AD to gain a critical mass to overcome the other agents. Figure 9 demonstrates that a sufficiently high proportion of AC effectively dooms cooperative outcomes.

Figure 9: AD dominates a nation of AC with insufficient TFT

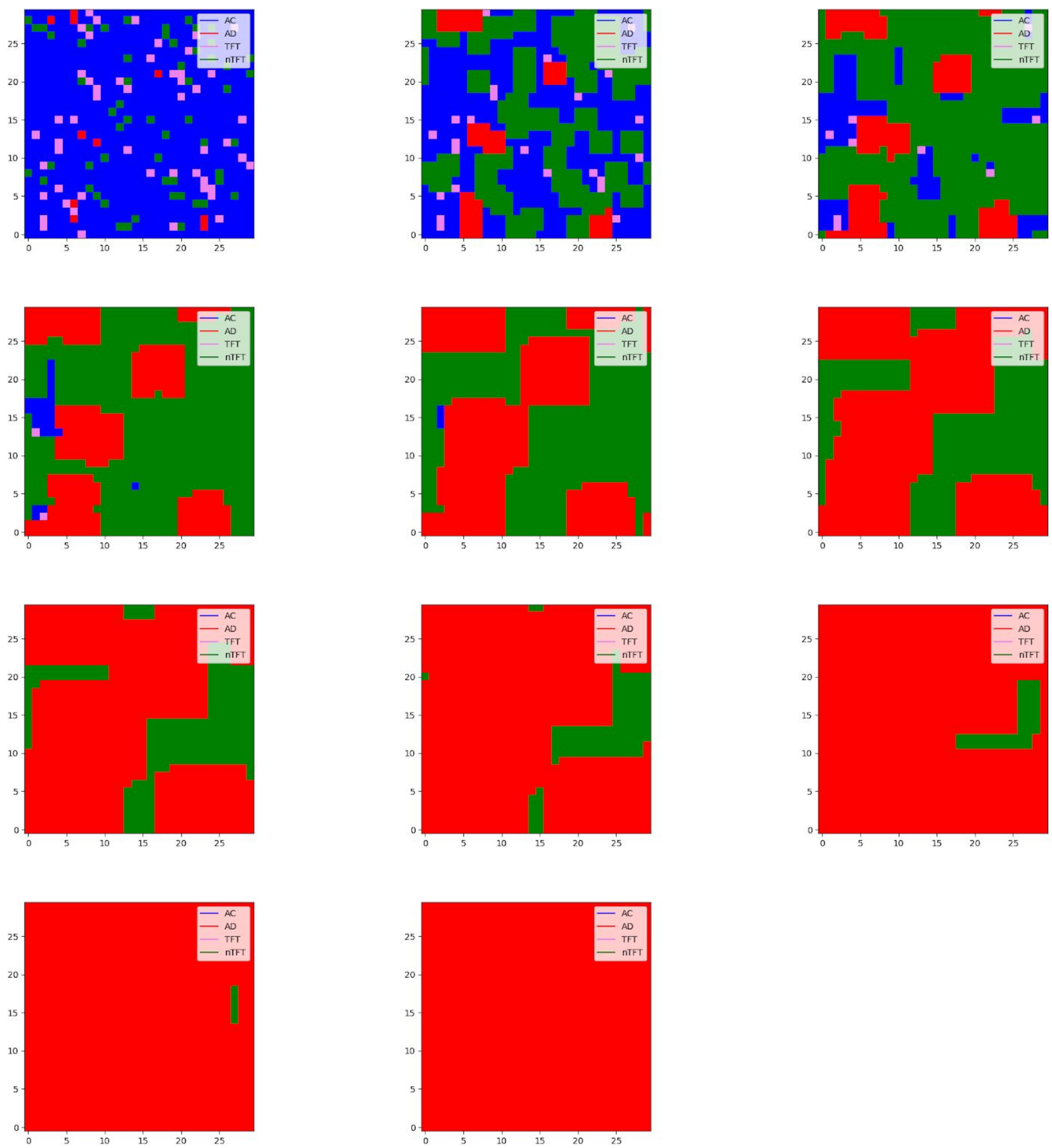
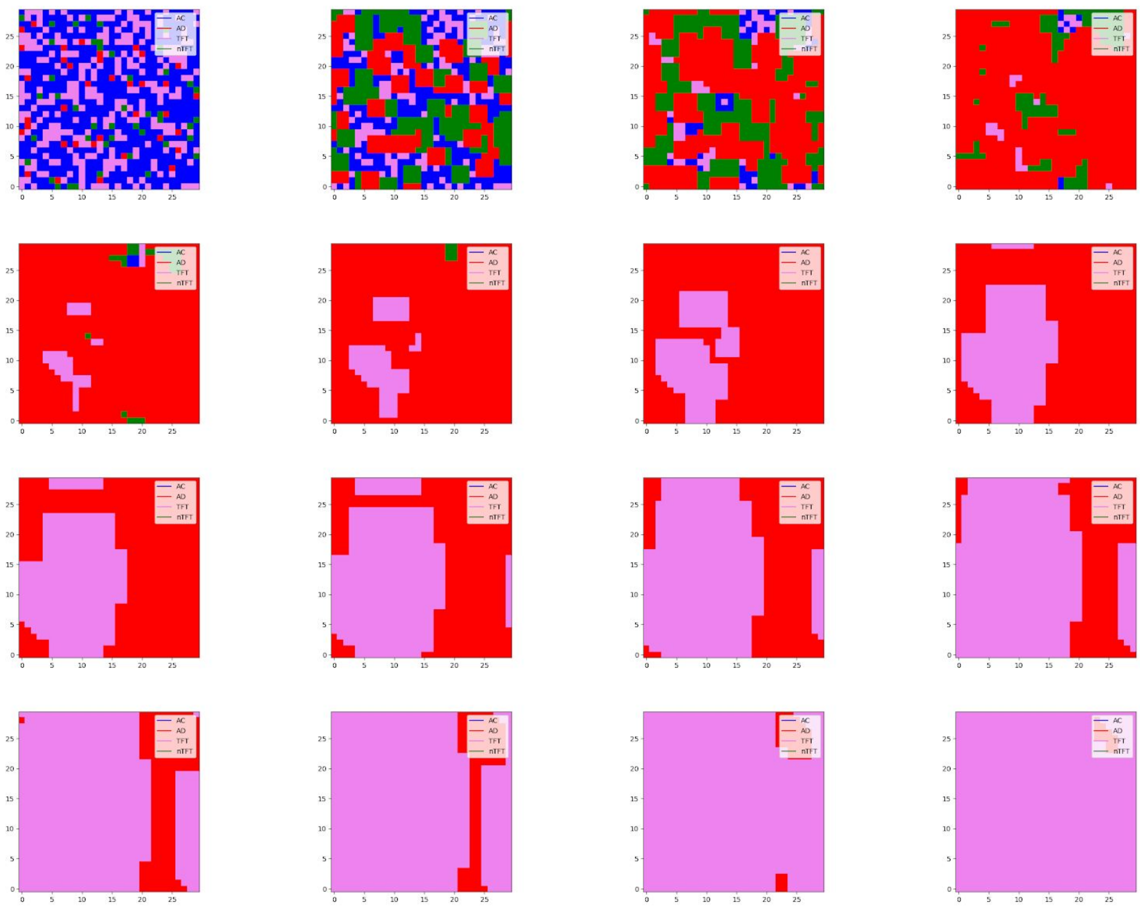


Figure 10 demonstrates that, if there exists a TFT cluster after the first several rounds[7], it can eventually overcome AD.

[7] Specifically, a cluster surrounded by AD

Figure 10: TFT usually overcomes AD if a TFT cluster survives early rounds

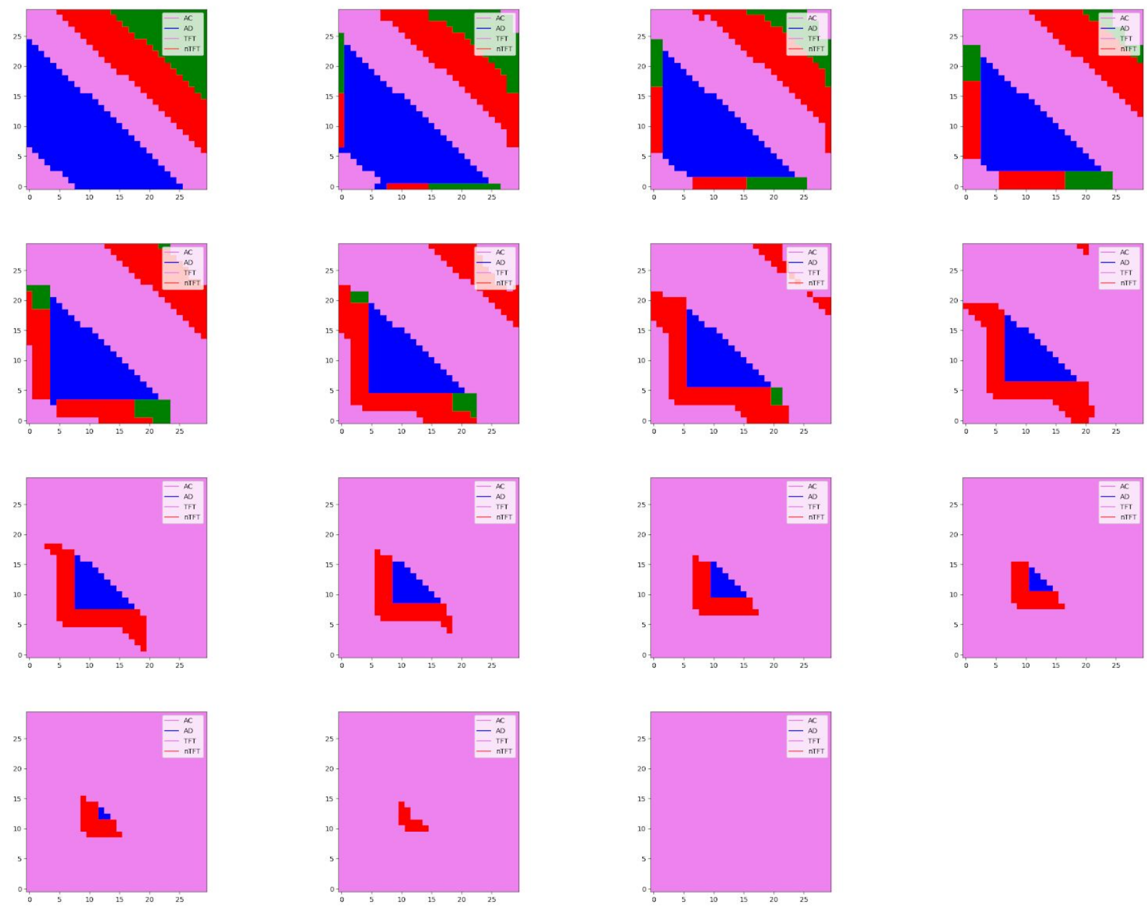


Finally, we know that AC and TFT is a valid, though rare equilibrium. One way for this to happen is for TFT to completely surround AC, effectively protecting it from the defectors.[8] However, Figure 11 elucidates why this outcome is so rare. For, even though AC started with a large contingent and is protected by TFT on multiple sides, it nevertheless succumbed to a sufficiently large breach. Meanwhile, TFT did nothing to fight these defector invaders until AC’s lamentable fate was sealed.

---

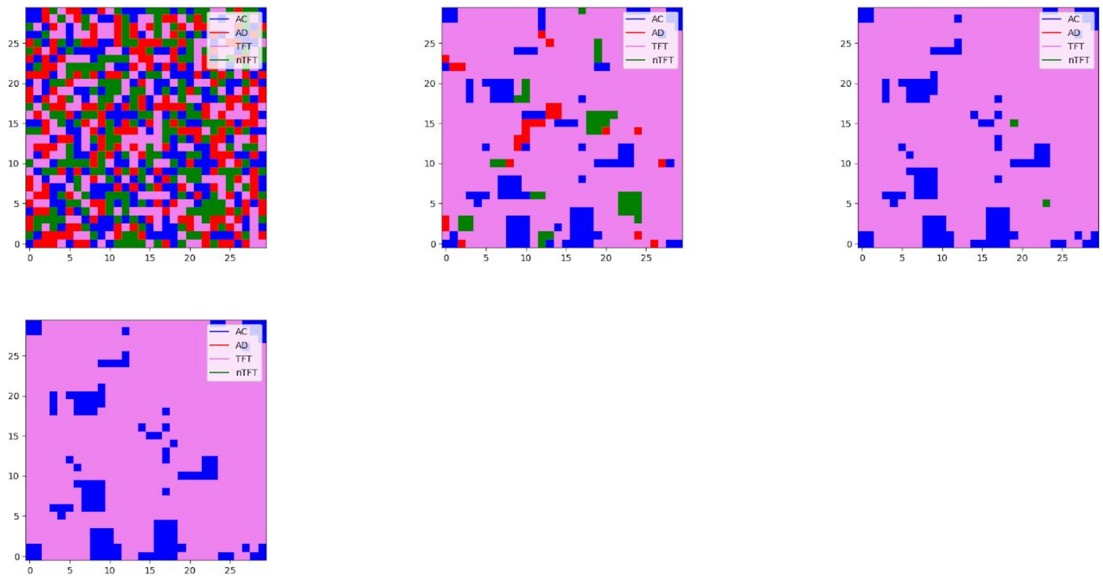
[8] Perhaps not unlike the Nephites and the Anti-Nephi-Lehies.

Figure 11: TFT will not proactively save AC



For the Stag Hunt, recall that cooperative agents are more likely to succeed. Indeed, Figure 12 demonstrates that convergence can happen very quickly.

Figure 12: Stag Hunt quickly converges to a cooperative outcome



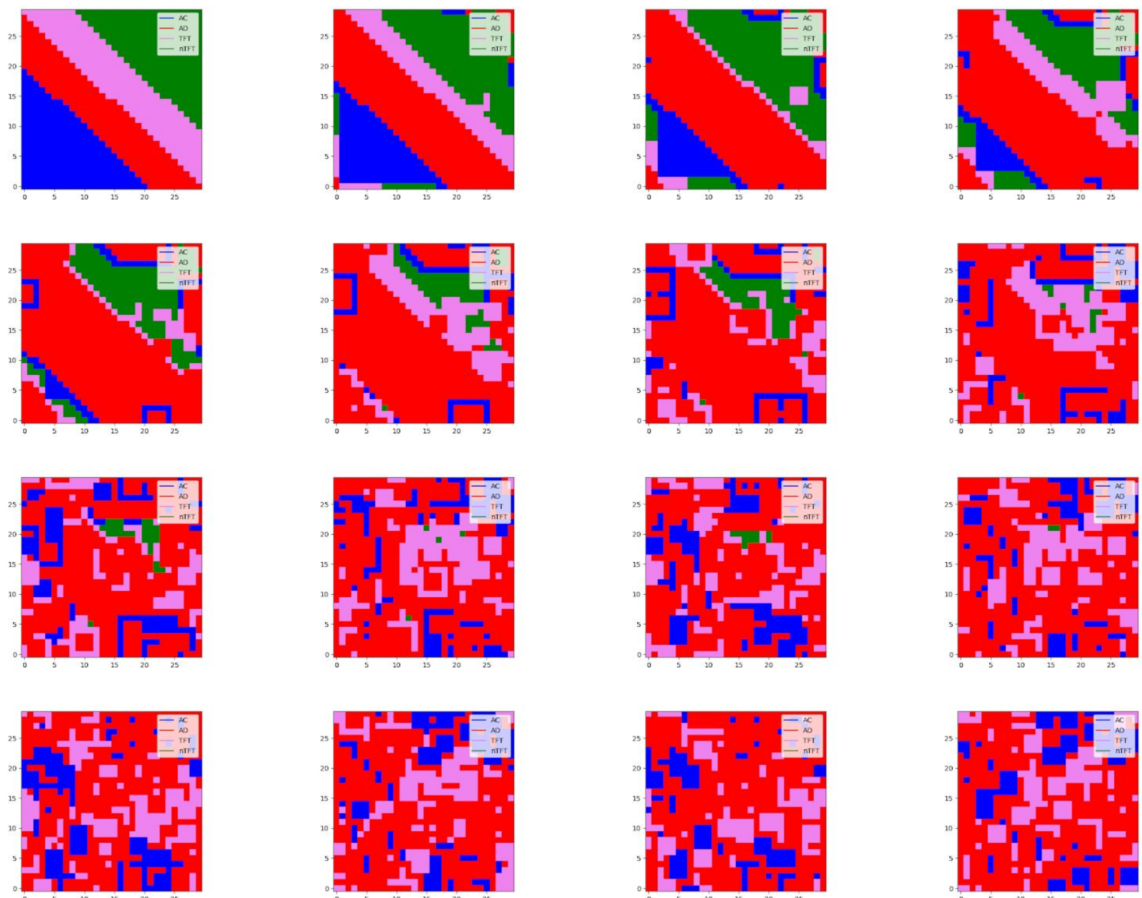
Notably, the Battle of the Sexes did not converge within 30 generations. As previously explained, no agent can cooperate with itself. Hence, rather than converging to one agent type, we observe agents cycling between two or three agent types.

Figure 13: Battle of the Sexes does not converge to one configuration

Game	Survivors	Observations	% of observations	Average initialization			
				% AC	% AD	% TFT	% nTFT
Battle of the Sexes	AC, AD, TFT	174	87%	27%	27%	20%	25%
	AD, TFT	25	13%	11%	17%	37%	34%
	AD, TFT, nTFT	1	1%	49%	0%	30%	21%

Figure 14 demonstrates this dynamic in action, with the later frames appearing to alternate between two configurations. Essentially, we have clusters of AD having low payoffs because they cannot coordinate with each other; because AC and TFT are comparatively successful when surrounded by AD, many ADs are replaced by TFT or AC in the next round. Coordination is thus restored, but because the AD's receive their preferred cooperative payoff, the recently converted TFT's and AC's largely convert back to AD.

FIgure 14: Battle of Sexes does not converge



Conclusion

In large measure, we observe that the replicator and imitator dynamics are similar. While the results of the imitator dynamic model can depend on how agents are initially clustered, it often converges to the same outcome as the replicator dynamic model.

There are a few areas for future work inspired by these experiments. In Battle of the Sexes, we saw the emergence of “symbiosis”, where agents depend on agents of other types for survival. A natural follow-on would be to design a game where teams of different agents types must coordinate with each other against other teams of agents to survive.

Another interesting feature, demonstrated in the Prisoner’s Dilemma, was that TFT was the sole survivor because it permitted AD to wipe out AC. Without AD, TFT could not take over without deviating from an “altruism-first” strategy. This raises important deontological questions about the morality of agents, e.g. can you design an agent that both wins and positions itself as morally superior?

Perhaps more interestingly is this idea that agents could be designed intentionally to succeed against some agents and succumb to others, with the intent of dominating globally rather than, e.g., doing as well as possible in each neighborhood.[9] Taking this a step further, we might have agents who can tactically alter their strategies based on their placement within a community[10]. For instance, perhaps a ring of meta-TFT agents surrounding a ring AC agents intentionally “surrender” to an army of AD, anticipating another wave of meta-TFT agents to come in later.[11]

Or perhaps we permit agent families to collectively alter their strategy between generations, based on their experience in the previous generation. This is a truly powerful idea; for, in any evolutionary system, no single agent is simultaneously best suited to every task, in every climate, against every agent; rather, the extent to which any agent will succeed in a dynamically changing environment must invariably be predicated on that agent’s ability to adapt.

---

[9] E.g., in a game of rock-paper-scissors with agents like “Always Rock”, “Always Scissors”, etc., a “paper-first” agent would do well to let the rock population take out scissors before making a play for dominance.

[10] But still without knowing precisely what agents they face, and still requiring they play the same strategy for all neighbors.

[11] We might still impose a rule where each agent must play the same strategy against all neighbors, but it has some knowledge of who those neighbors are.