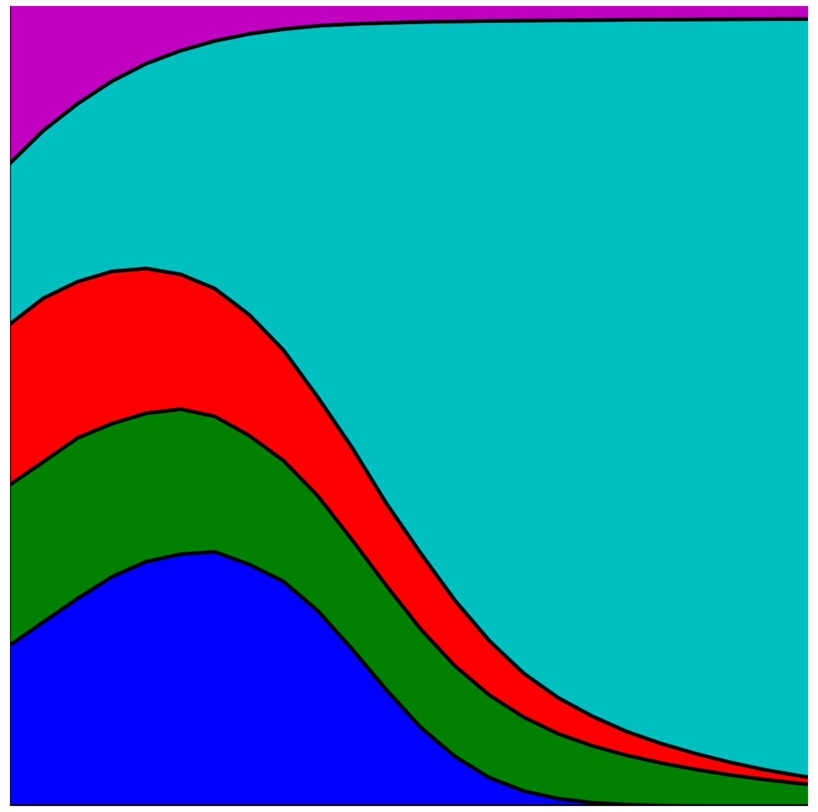
A computational approach to cooperation

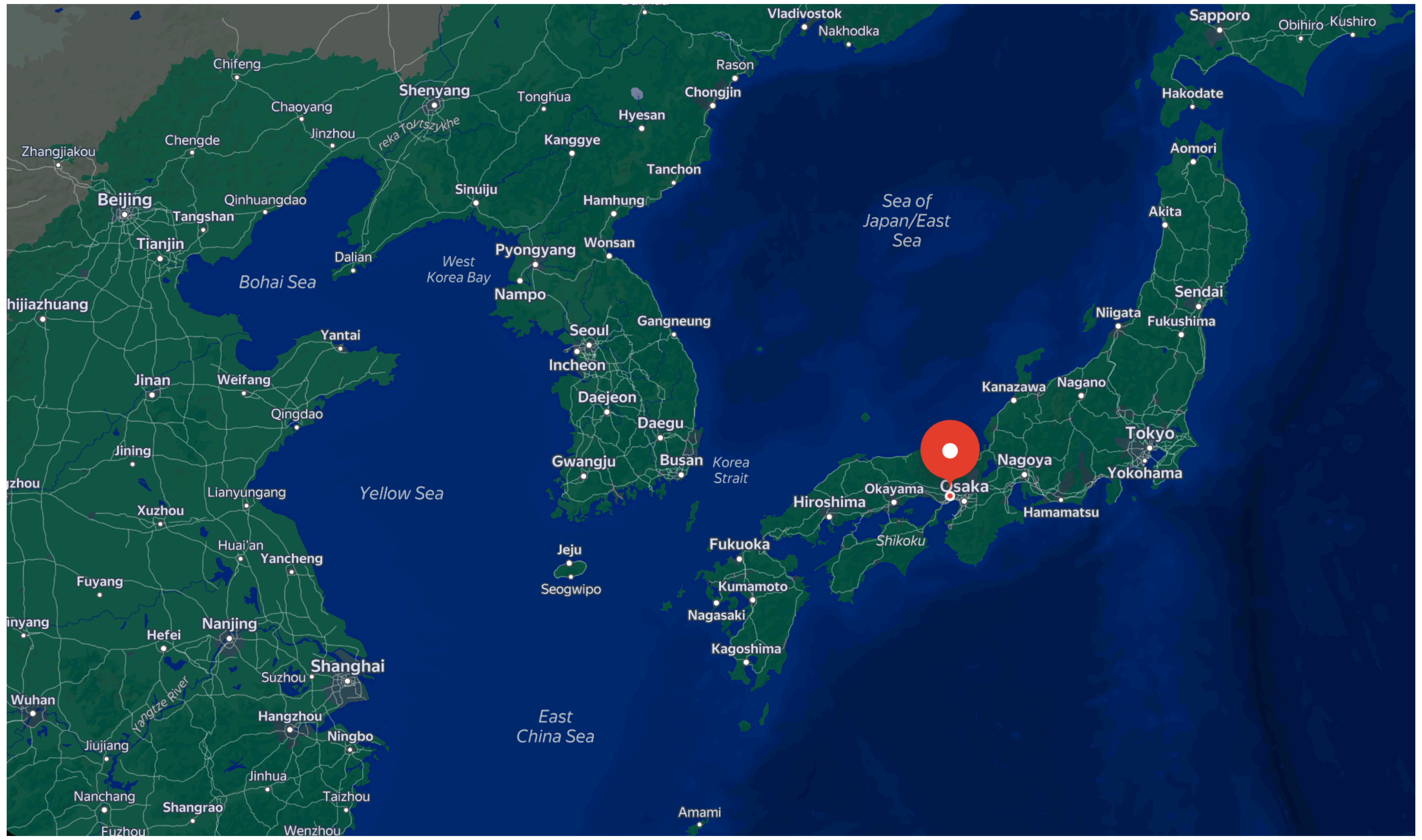
Game Theory—Summer School—Kyiv

Nikoleta E. Glynatsi

RIKEN

R-CCS





Material

[Game-theory-summer-school-Kyiv](https://github.com/Nikoleta-v3/Game-theory-summer-school-Kyiv)

https://github.com/Nikoleta-v3/Game-theory-summer-school-Kyiv

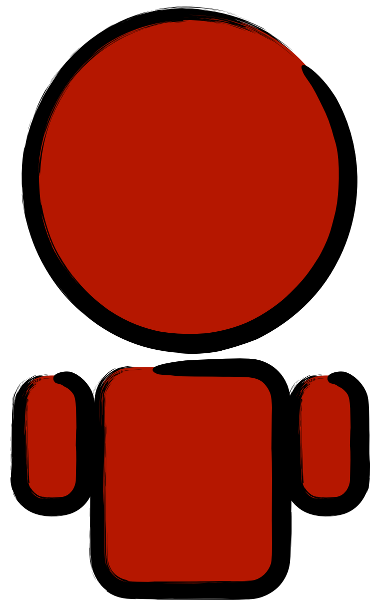
Contact

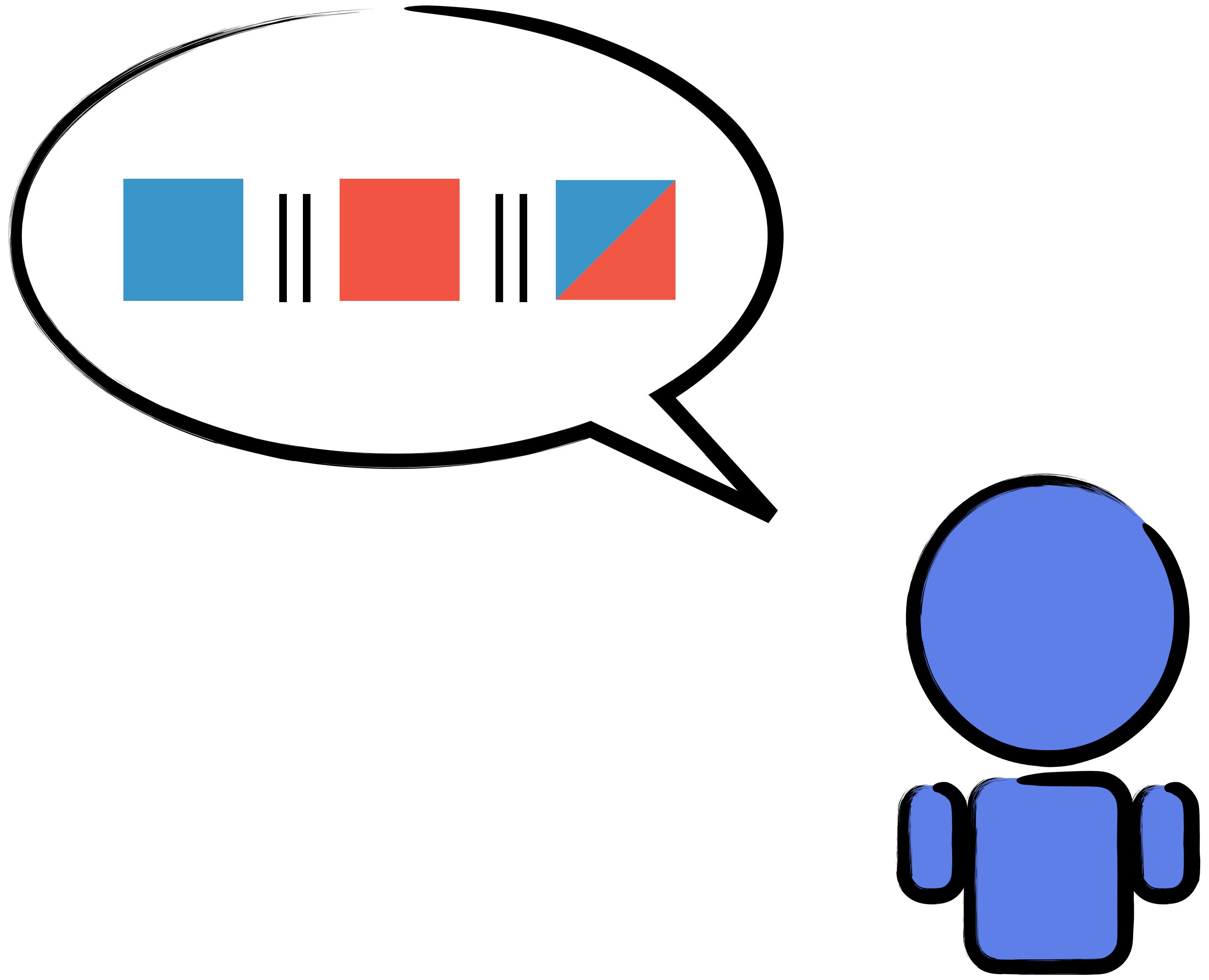
Nikoleta-v3

nikoleta.glynatsi@riken.jp

1. Introduction to repeated games
2. Computer tournaments
3. Axelrod-Python
4. Training Strategies
5. Properties of successful strategies

Introduction to repeated games



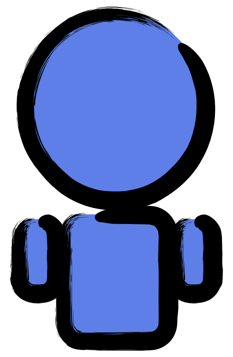
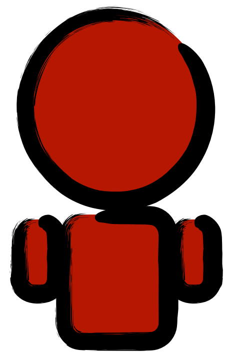


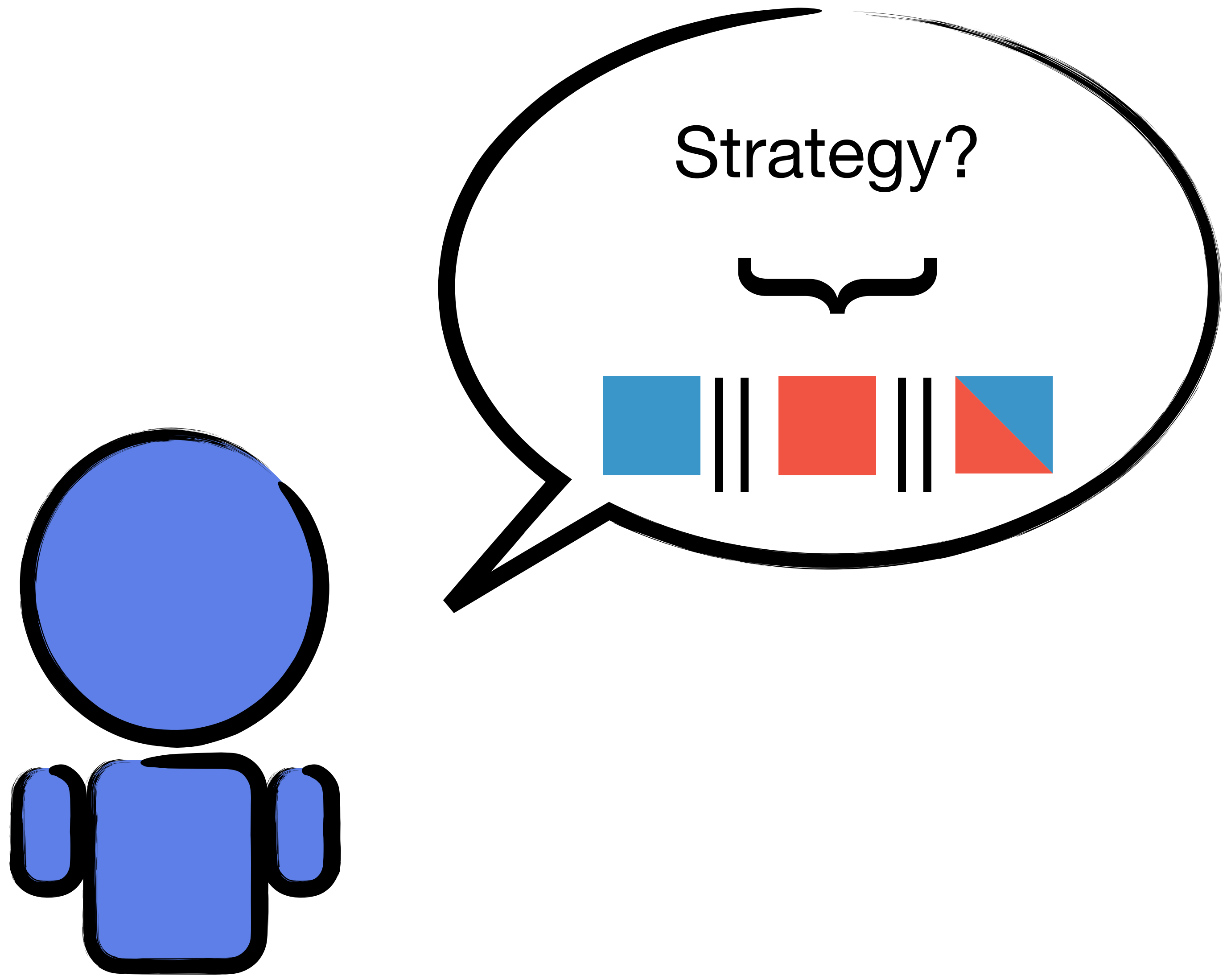
$$\begin{array}{c c} & \begin{array}{cc} C & D \end{array} \\ \begin{array}{c} C \\ D \end{array} & \begin{pmatrix} 3 & 0 \\ 5 & 1 \end{pmatrix} \end{array}$$

Nash Equilibrium

$$T > R > P > S$$

Strategies

	1	2	3		$n-1$	n	$n+1$	
	$\begin{array}{c c}& C \quad D\\ \begin{matrix}C\\D\end{matrix} & \begin{pmatrix} r & s \\ t & p \end{pmatrix}\end{array}$	$\begin{array}{c c}& C \quad D\\ \begin{matrix}C\\D\end{matrix} & \begin{pmatrix} r & s \\ t & p \end{pmatrix}\end{array}$	$\begin{array}{c c}& C \quad D\\ \begin{matrix}C\\D\end{matrix} & \begin{pmatrix} r & s \\ t & p \end{pmatrix}\end{array}$	$\cdots$	$\begin{array}{c c}& C \quad D\\ \begin{matrix}C\\D\end{matrix} & \begin{pmatrix} r & s \\ t & p \end{pmatrix}\end{array}$	$\begin{array}{c c}& C \quad D\\ \begin{matrix}C\\D\end{matrix} & \begin{pmatrix} r & s \\ t & p \end{pmatrix}\end{array}$	$\begin{array}{c c}& C \quad D\\ \begin{matrix}C\\D\end{matrix} & \begin{pmatrix} r & s \\ t & p \end{pmatrix}\end{array}$	∞
	C	D	C	$\cdots$	C	C	C/D	
	D	C	C	$\cdots$	D	C		



Computer tournaments

What strategy should Blue play to be successful?

Which is the best strategy in the repeated prisoner's dilemma

Robert Axelrod

1 Introduction to repeated games

2 Computer tournaments

3 Axelrod-Python

4 Training Strategies

5 Properties of successful strategies

Axelrod’s First

RULES

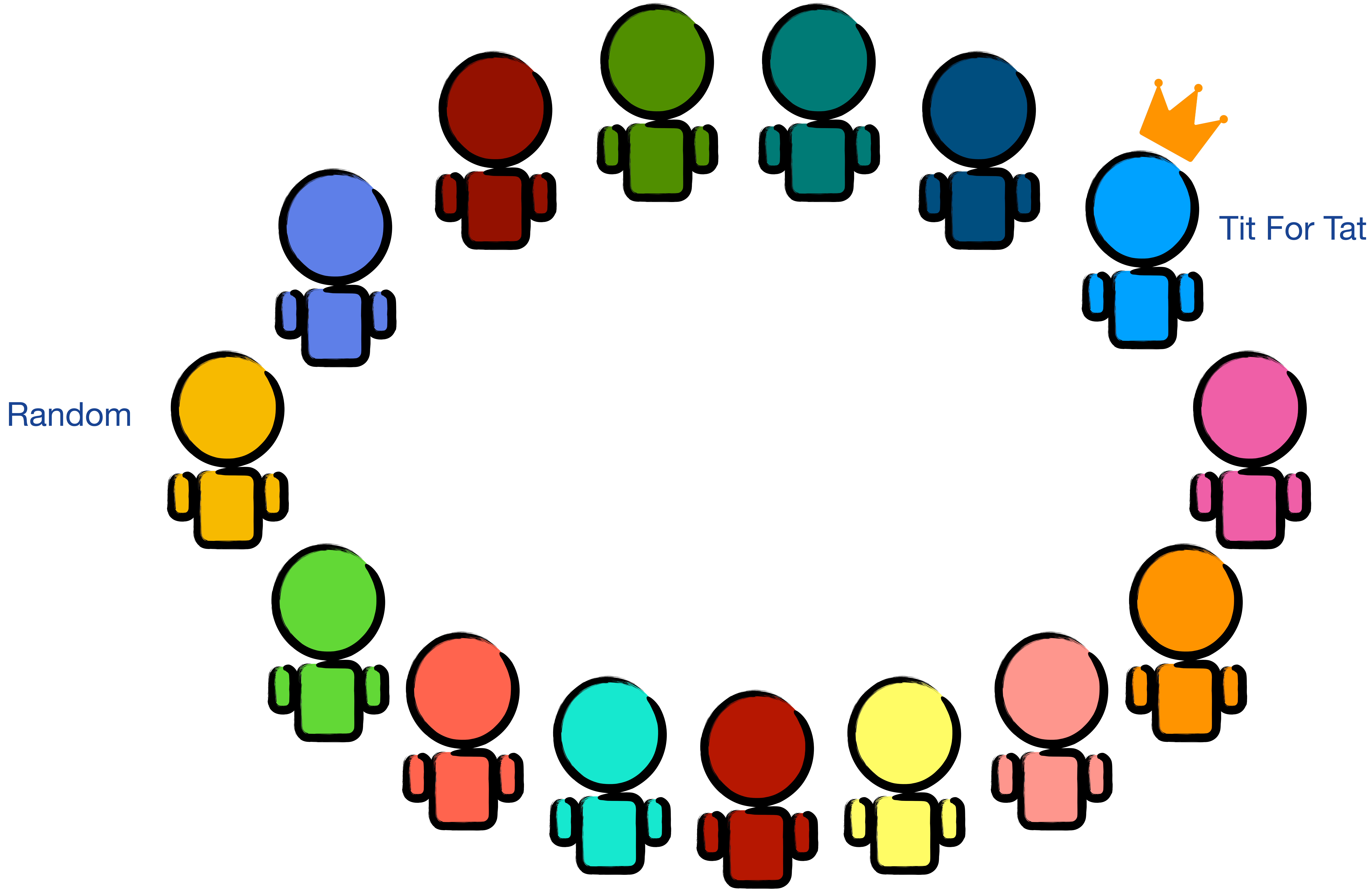
- Strategies are submitted as computer code.
- Each strategy plays a match against every other strategy, a copy of itself, and the Random strategy.
- Each match consists of 200 rounds.
- The tournament is repeated five times.
- The strategy with the highest average payoff across all matches is declared the winner.

Effective Choice in the Prisoner’s Dilemma

ROBERT AXELROD
Institute of Public Policy Studies
University of Michigan

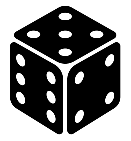
This is a “primer” on how to play the iterated Prisoner’s Dilemma game effectively. Existing research approaches offer the participant limited help in understanding how to cope effectively with such interactions. To gain a deeper understanding of how to be effective in such a partially competitive and partially cooperative environment, a computer tournament was conducted for the iterated Prisoner’s Dilemma. Decision rules were submitted by entrants who were recruited primarily from experts in game theory from a variety of disciplines: psychology, political science, economics, sociology, and mathematics. The results of the tournament demonstrate that there are subtle reasons for an individualistic pragmatist to cooperate as long as the other side does, to be somewhat forgiving, and to be optimistic about the other side’s responsiveness.

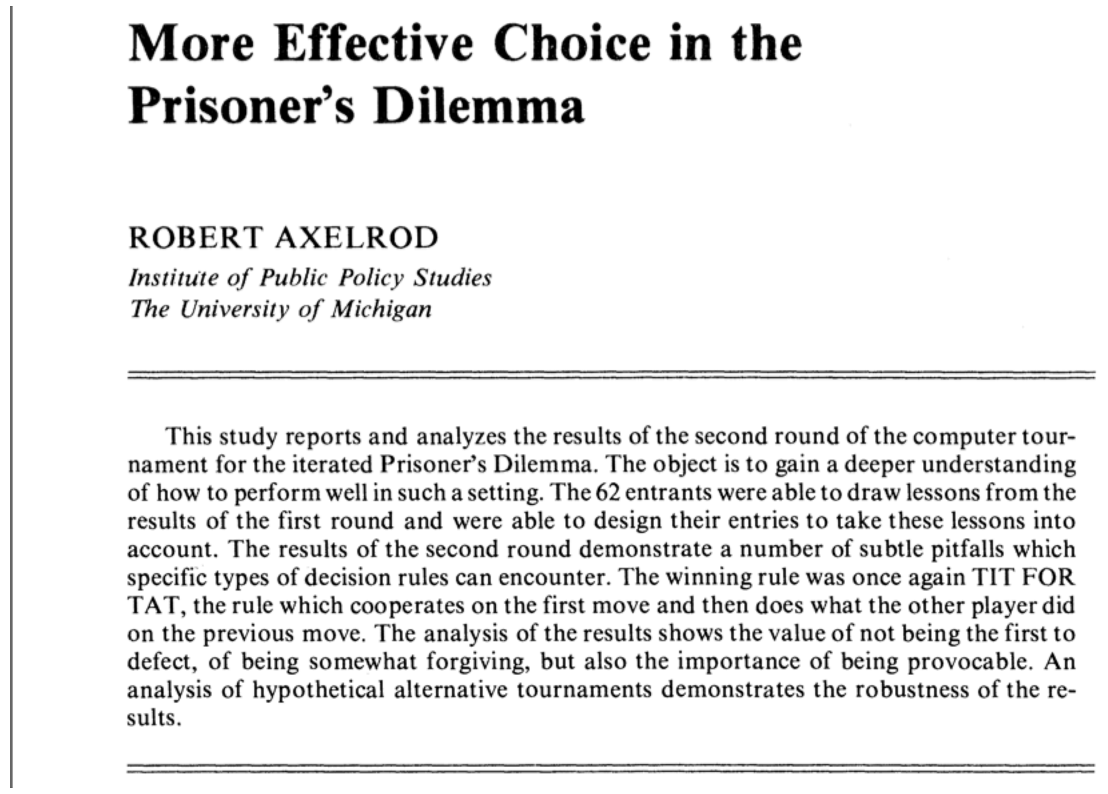
Axelrod’s First



Axelrod’s Second

SECOND TOURNAMENT RULES

- Strategies are submitted as computer code.
- Each strategy plays a match against every other strategy, a copy of itself, and the Random strategy.
- Each match consists of  rounds.
- The tournament is repeated five times.
- The strategy with the highest average payoff across all matches is declared the winner.



More Effective Choice in the Prisoner’s Dilemma

ROBERT AXELROD
Institute of Public Policy Studies
The University of Michigan

This study reports and analyzes the results of the second round of the computer tournament for the iterated Prisoner’s Dilemma. The object is to gain a deeper understanding of how to perform well in such a setting. The 62 entrants were able to draw lessons from the results of the first round and were able to design their entries to take these lessons into account. The results of the second round demonstrate a number of subtle pitfalls which specific types of decision rules can encounter. The winning rule was once again TIT FOR TAT, the rule which cooperates on the first move and then does what the other player did on the previous move. The analysis of the results shows the value of not being the first to defect, of being somewhat forgiving, but also the importance of being provocable. An analysis of hypothetical alternative tournaments demonstrates the robustness of the results.

Axelrod's Second

Other Strategies

- Computer Tournaments
- Evolutionary Dynamics
- Two players' interactions

Our Meeting With Gradual: A *Good* Strategy For The Iterated Prisoner's Dilemma

Bruno Beaufils, Jean-Paul Delahaye and Philippe Mathieu
Université des Sciences et Technologies de Lille
Laboratoire d'Informatique Fondamentale de Lille – U.R.A. 369 C.N.R.S.
U.F.R. d'I.E.E.A. Bât. M3 – F-59655 Villeneuve d'Ascq Cedex
e-mail: {beaufils, mathieu, delahaye}@lifl.fr

Abstract

In this paper, after a short return to the description of the classical version of the Iterated Prisoner's Dilemma and its application to the study of cooperation, we present a new strategy we have found named *gradual*, which outperforms the *tit-for-tat* strategy, on which are based a lot of works in the Theory of Cooperation. Since no pure strategy is evolutionarily stable in the IPD, we cannot give a mathematical proof of the absolute superiority of *gradual*, but we present a convergent set of facts that must be viewed as strong experimental evidences of

In section 3 we try to improve the strength of this strategy by using a genetic algorithm, on a genotype we have created and which includes lots of well-known strategies (in fact our genotype can cover more than 8×10^{15} strategies). We present our ideas on a tree representation of the strategies space and finally we propose a new view of evolution of cooperation in which complexity plays a major role.

In the last sections we describe our results and we discuss about the natural behavior of this strategy and its good robustness in ecological competitions.

1.1 IPD and Artificial Life

A strategy of win–stay, lose–shift that outperforms tit-for-tat in the Prisoner's Dilemma game

Martin Nowak* & Karl Sigmund†

*Department of Zoology, University of Oxford, South Parks Road, Oxford OX1 3PS, UK
†Institut für Mathematik, Universität Wien, Strudlhofgasse 4, A-1090 Vienna, Austria

The Prisoner's Dilemma is the leading metaphor for the evolution of cooperative behaviour in populations of selfish agents, especially since the well-known computer tournaments of Axelrod[1] and their application to biological communities[2,3]. In Axelrod's simulations, the simple strategy tit-for-tat did outstandingly well and subsequently became the major paradigm for reciprocal altruism[4 12]. Here we present extended evolutionary simulations of heterogeneous ensembles of probabilistic strategies including mutation and selection, and report the unexpected success of another protagonist: Pavlov. This strategy is as simple as tit-for-tat and embodies the fundamental behavioural mechanism win–stay, lose–shift, which seems to be a widespread rule[13]. Pavlov's success is based on two important advantages over tit-for-tat: it can correct occasional mistakes and exploit unconditional cooperators. This second feature prevents Pavlov populations from being undermined by unconditional cooperators, which in turn invite defectors. Pavlov seems to be more robust than tit-for-tat, suggesting that cooperative behaviour in natural situations may often be based on win–stay, lose–shift.

The conspicuous success of the tit-for-tat (TFT) strategy (start with a C, and then use your co-players previous move) relies in part on the clinical neatness of a deterministic cyber-world. In natural populations, errors occur[7,12]. TFT suffers from stochastic perturbations in two ways: (1) a TFT population can be 'softened up' by random drift introducing unconditional cooperators, which allow exploiters to grow (TFT is not an evolutionarily stable strategy[15,16]); and (2) occasional mistakes between two TFT players cause long runs of mutual backbiting. (Such mistakes abound in real life: even humans are apt to vent frustrations upon innocent bystanders.)

Within the restricted world of strategies reacting only to the co-players previous move, TFT has a very important, but transitory role: in small clusters, it can invade populations of defectors, but then bows out to a related strategy, 'generous tit for tat' (GTFT), which cooperates after a co-player's C, but also with a certain probability after a D[9].

But as soon as one admits strategies which take into account the moves of both players in the previous round, evolution becomes much less transparent[17]. We first conjectured that GTFT (or variants thereof) would win the day, but are forced to admit, after extensive simulations, that the strategy Pavlov did much better in the long run. A Pavlov player cooperates if and only if both players opted for the same alternative in the previous move. The name[18] stems from the fact that this strategy embodies an almost reflex-like response to the payoff: it repeats its former move if it was rewarded by R or T points, but switches behaviour if it was punished by receiving only P or S points. This strategy, which went by the name of 'simpleton'[19], fares poorly against inveterate defectors: in every second round, it switches to cooperation. It cannot gain a foothold in a defector's world; defectors have to be invaded by other strategies, like TFT[9]. But Pavlov has two important advantages over TFT:

OPEN

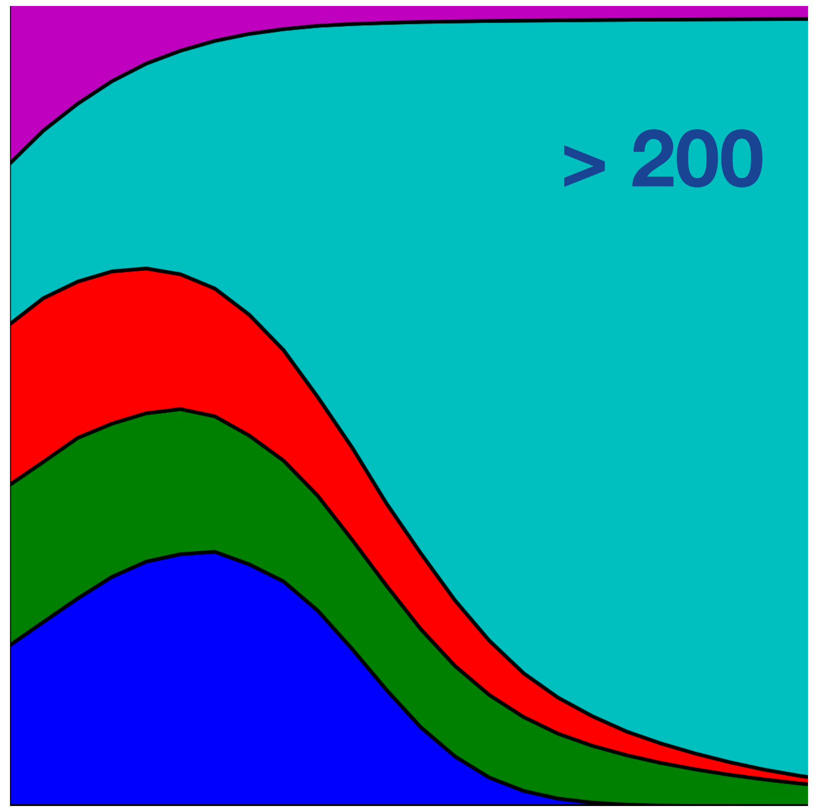
Five rules for friendly rivalry in direct reciprocity

Yohsuke Murase[1] & Seung Ki Baek[2]✉

Direct reciprocity is one of the key mechanisms accounting for cooperation in our social life. According to recent understanding, most of classical strategies for direct reciprocity fall into one of two classes, 'partners' or 'rivals'. A 'partner' is a generous strategy achieving mutual cooperation, and a 'rival' never lets the co-player become better off. They have different working conditions: For example, partners show good performance in a large population, whereas rivals do in head-to-head matches. By means of exhaustive enumeration, we demonstrate the existence of strategies that act as both partners and rivals. Among them, we focus on a human-interpretable strategy, named 'CAPRI' after its five characteristic ingredients, i.e., cooperate, accept, punish, recover, and defect otherwise. Our evolutionary simulation shows excellent performance of CAPRI in a broad range of environmental conditions.

Axelrod-Python

https://github.com/Axelrod-Python/Axelrod



https://axelrod.readthedocs.io/en/stable/

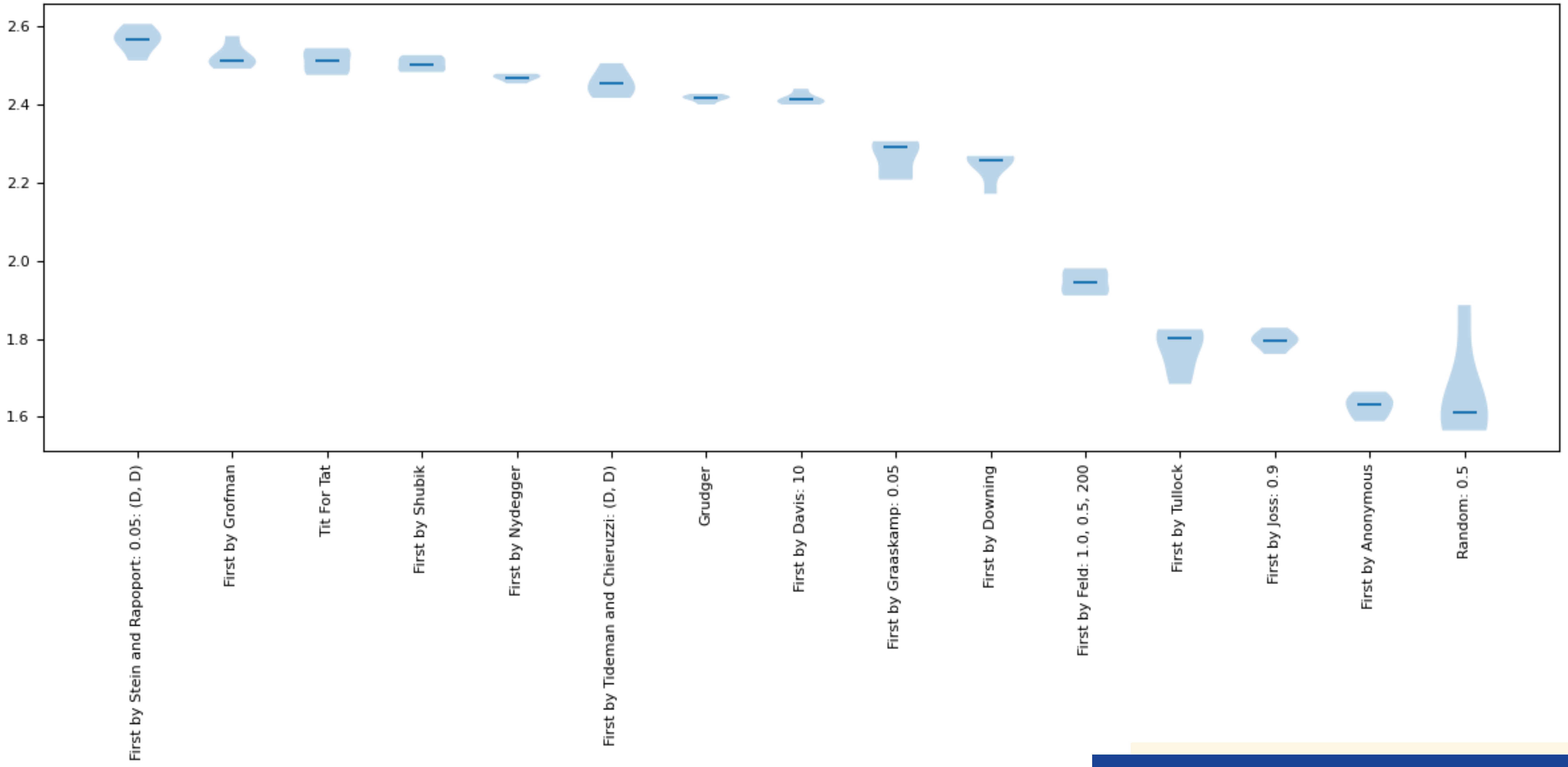
Help desk

https://discord.gg/XCMuZDhQ

1. Introduction to repeated games
2. Computer tournaments
3. Axelrod-Python
4. Training Strategies
5. Properties of successful strategies

Match

Tournament

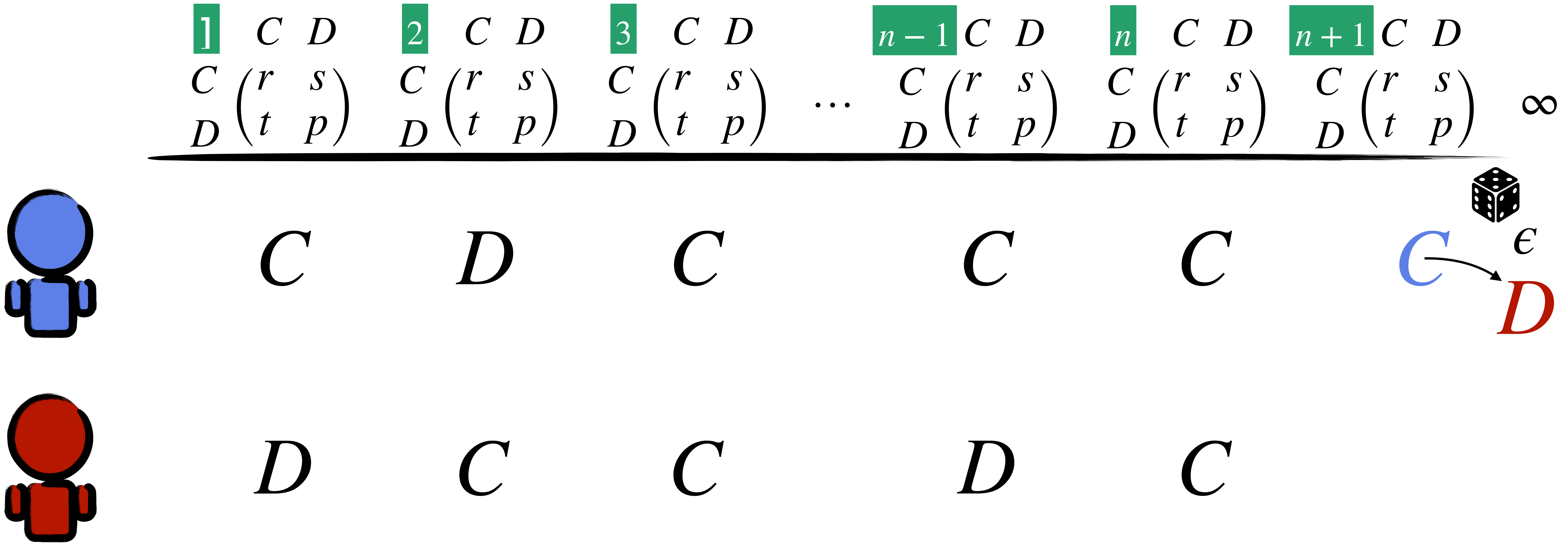


Can we replicate the original result ?

- Axelrod's paper in the repository
- Notebook in the repository
- Read about the first tournament here: https://axelrod.readthedocs.io/en/stable/discussion/overview_of_past_tournaments.html#axelrod-s-first-tournament

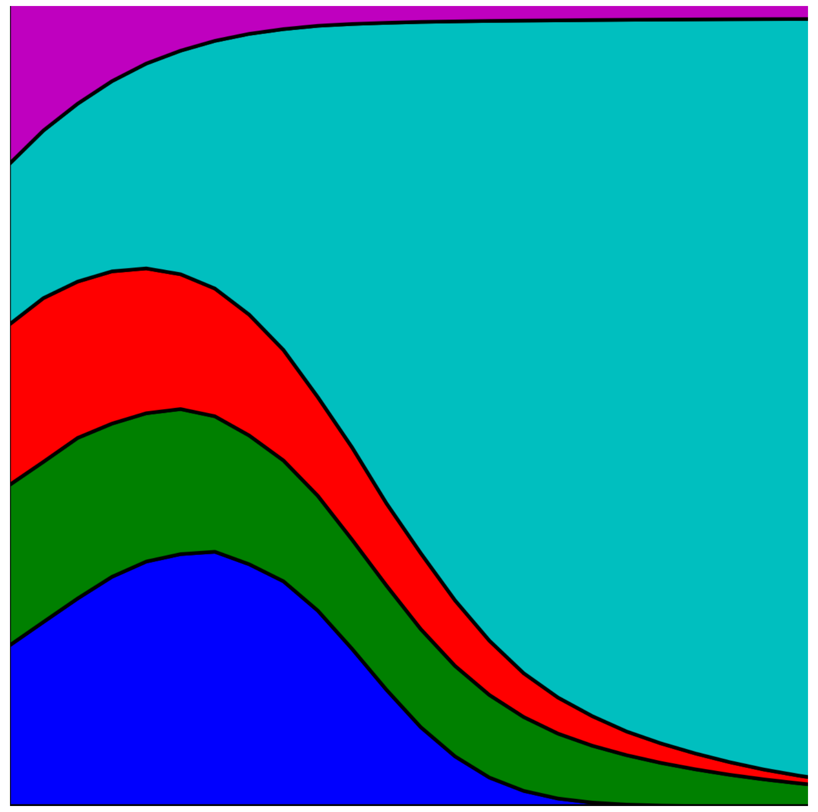
Probabilistic Ending

Noise



Axelrod-Python

https://github.com/Axelrod-Python/Axelrod



https://axelrod.readthedocs.io/en/stable/

Help desk

https://discord.gg/XCMuZDhQ

1. Introduction to repeated games
2. Computer tournaments
3. Axelrod-Python
4. Training Strategies
5. Properties of successful strategies

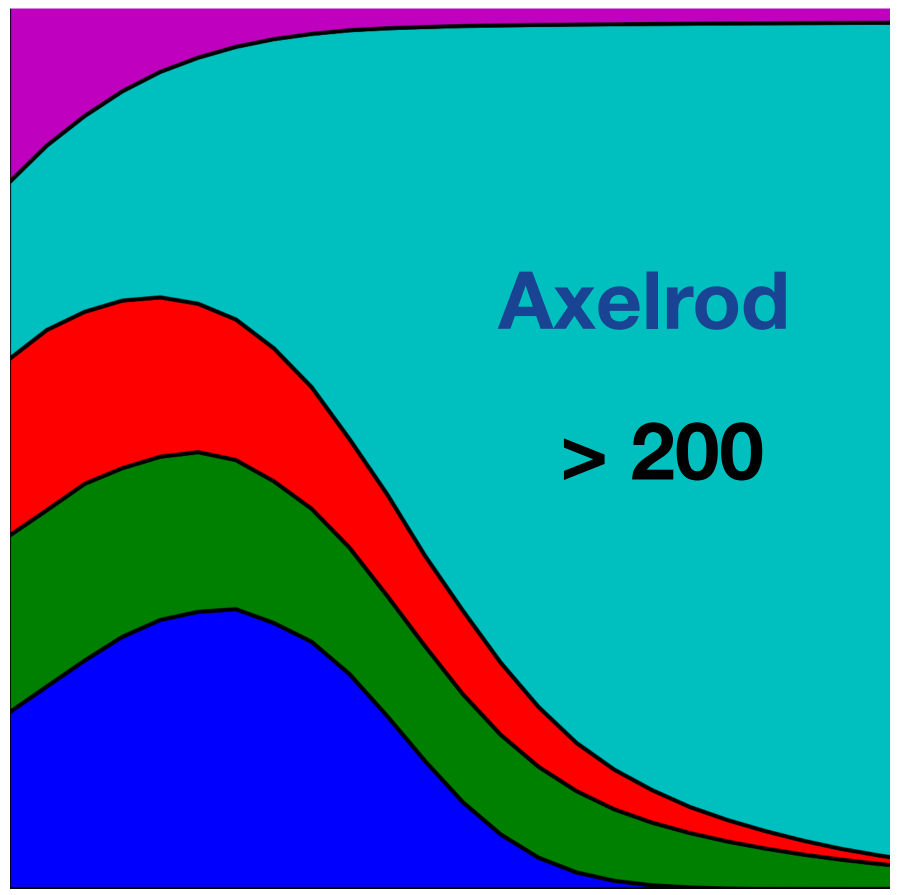
Training Strategies

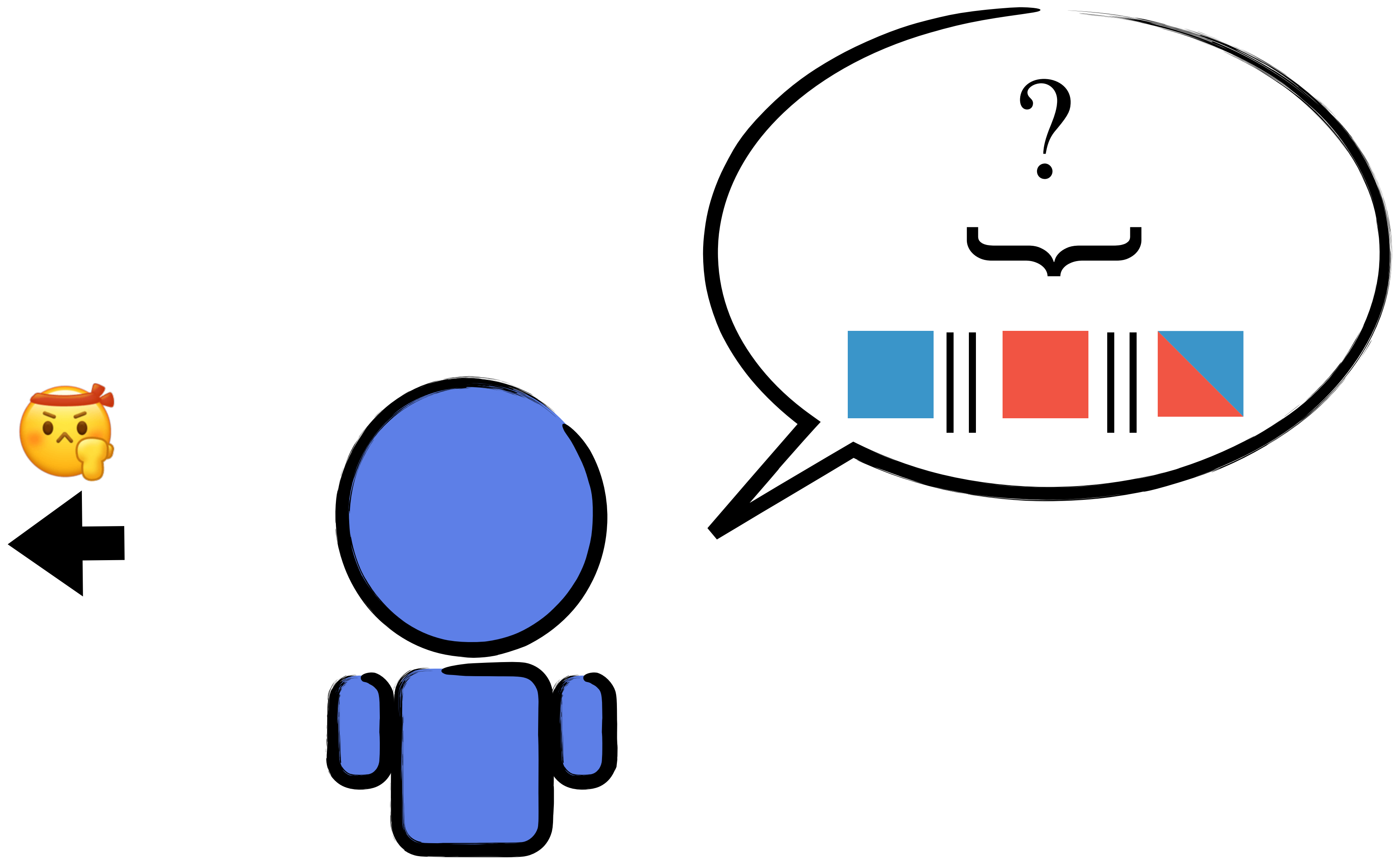




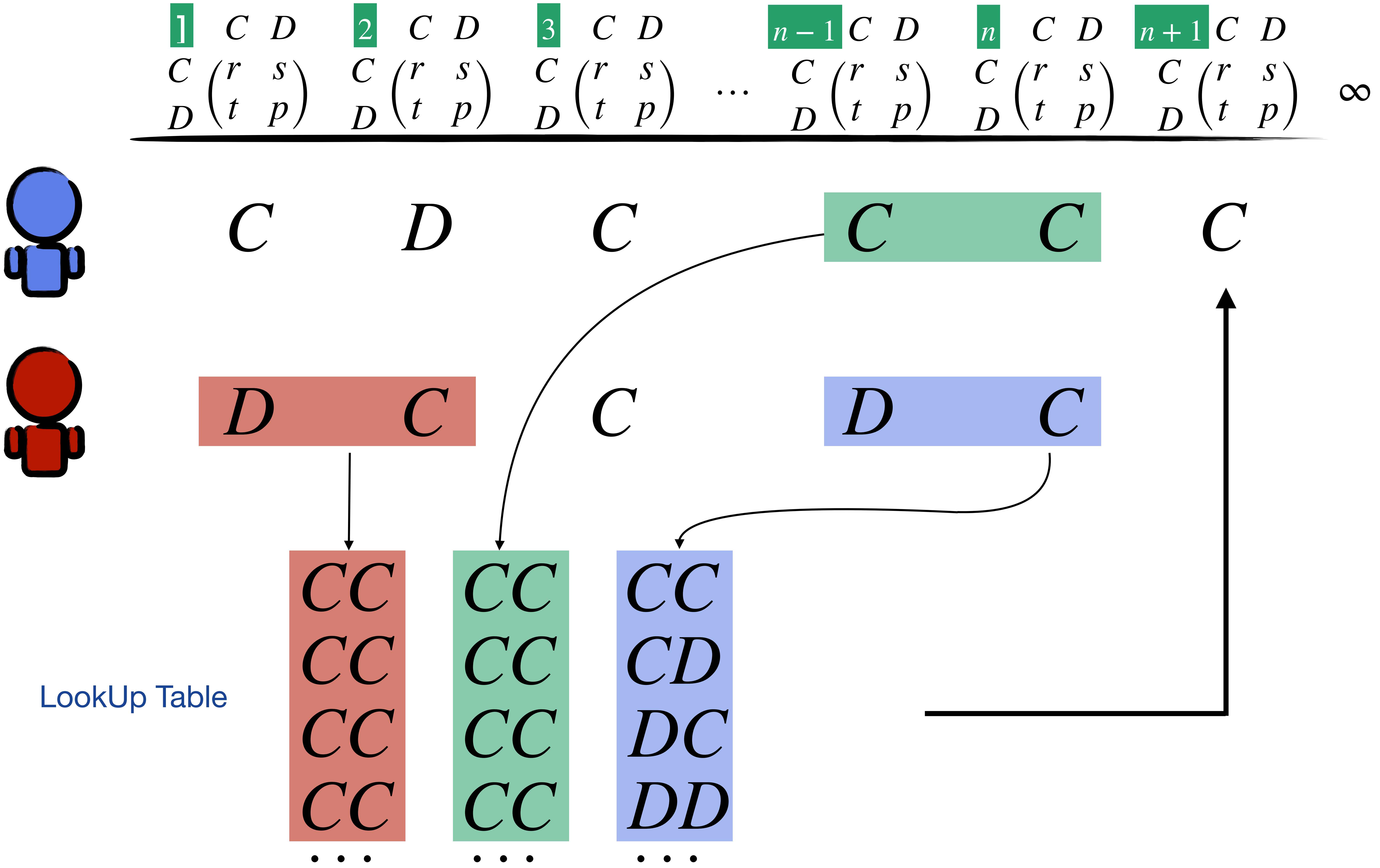


Training Strategies





Representing Strategies



Reinforcement learning produces dominant strategies for the Iterated Prisoner's Dilemma

Top performing strategies in a tournament with over 200 strategies.

	mean	std	min	5%	25%	50%	75%	95%	max
EvolvedLookerUp2_2_2*	2.955	0.010	2.915	2.937	2.948	2.956	2.963	2.971	2.989
Evolved HMM 5*	2.954	0.014	2.903	2.931	2.945	2.954	2.964	2.977	3.007
Evolved FSM 16*	2.952	0.013	2.900	2.930	2.943	2.953	2.962	2.973	2.993
PSO Gambler 2_2_2*	2.938	0.013	2.884	2.914	2.930	2.940	2.948	2.957	2.972
Evolved FSM 16 Noise 05*	2.919	0.013	2.874	2.898	2.910	2.919	2.928	2.939	2.965
PSO Gambler 1_1_1*	2.912	0.023	2.805	2.874	2.896	2.912	2.928	2.950	3.012
Evolved ANN 5*	2.912	0.010	2.871	2.894	2.905	2.912	2.919	2.928	2.945
Evolved FSM 4*	2.910	0.012	2.867	2.889	2.901	2.910	2.918	2.929	2.943
Evolved ANN*	2.907	0.010	2.865	2.890	2.900	2.908	2.914	2.923	2.942
PSO Gambler Mem1*	2.901	0.025	2.783	2.858	2.884	2.901	2.919	2.942	2.994
Evolved ANN 5 Noise 05*	2.864	0.008	2.830	2.850	2.858	2.865	2.870	2.877	2.891
DBS	2.857	0.009	2.823	2.842	2.851	2.857	2.863	2.872	2.899
Winner12	2.849	0.008	2.820	2.836	2.844	2.850	2.855	2.862	2.874
Fool Me Once	2.844	0.008	2.818	2.830	2.838	2.844	2.850	2.857	2.882
Omega TFT: 3, 8	2.841	0.011	2.800	2.822	2.833	2.841	2.849	2.859	2.882

https://doi.org/10.1371/journal.pone.0188046.t002

Training Strategies

- Understand the trained strategy
- Develop a more advanced or efficient reinforcement learning algorithm
- Explore better ways to represent strategies (e.g., finite state automata)

1 Introduction to repeated games

2 Computer tournaments

3 Axelrod-Python

4 Training Strategies

5 Properties of successful strategies

Properties of successful strategies

https://doi.org/10.1371/journal.pcbi.1012644






What strategy should Blue play to be successful?



What does it mean successful?

Performing well at a variety of tournaments.

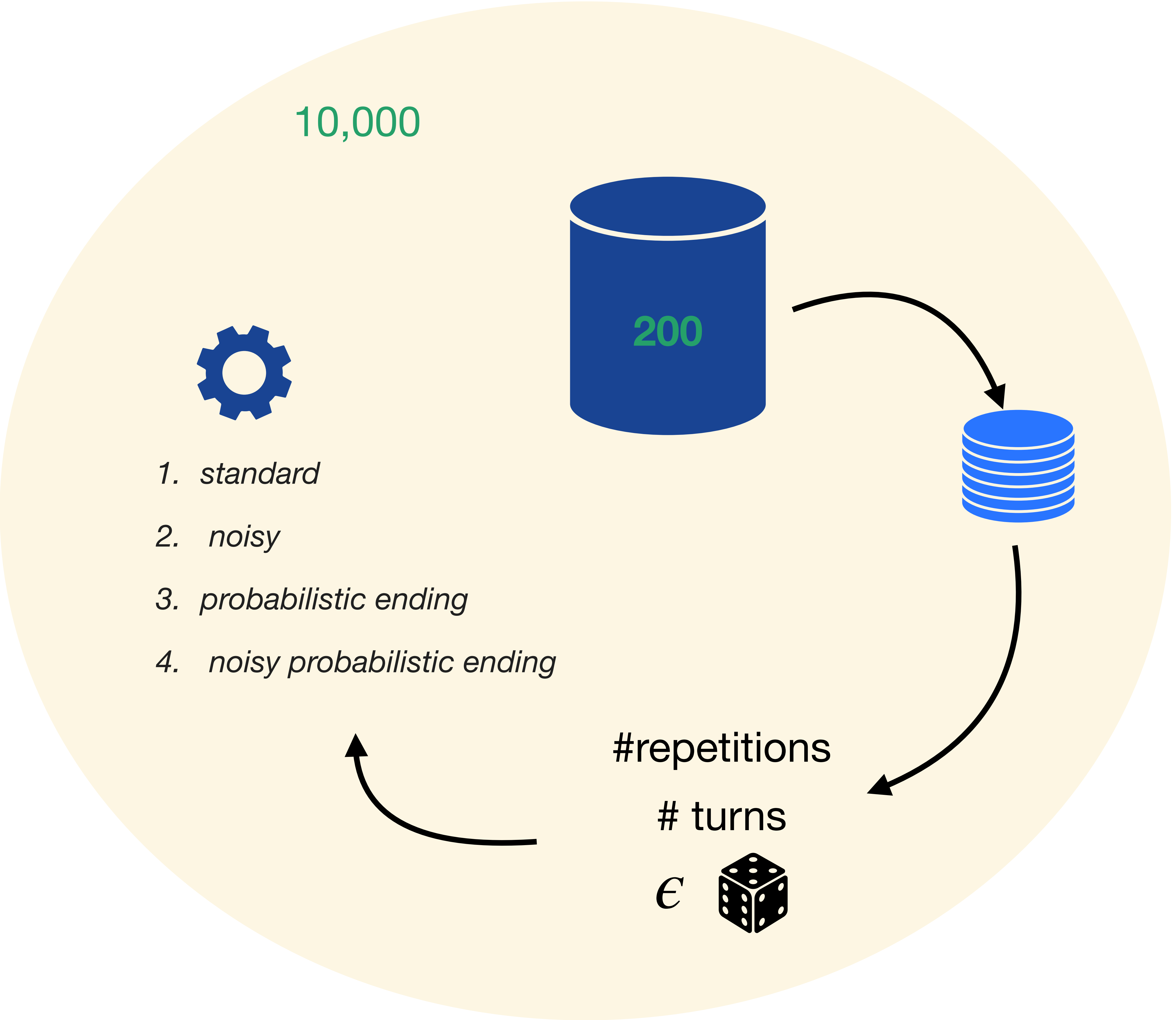
	mean	std	min	5%	25%	50%	75%	95%	max
EvolvedLookerUp2_2_2*	2.955	0.010	2.915	2.937	2.948	2.956	2.963	2.971	2.989
Evolved HMM 5*	2.954	0.014	2.903	2.931	2.945	2.954	2.964	2.977	3.007
Evolved FSM 16*	2.952	0.013	2.900	2.930	2.943	2.953	2.962	2.973	2.993
PSO Gambler 2_2_2*	2.938	0.013	2.884	2.914	2.930	2.940	2.948	2.957	2.972
Evolved FSM 16 Noise 05*	2.919	0.013	2.874	2.898	2.910	2.919	2.928	2.939	2.965
PSO Gambler 1_1_1*	2.912	0.023	2.805	2.874	2.896	2.912	2.928	2.950	3.012
Evolved ANN 5*	2.912	0.010	2.871	2.894	2.905	2.912	2.919	2.928	2.945
Evolved FSM 4*	2.910	0.012	2.867	2.889	2.901	2.910	2.918	2.929	2.943
Evolved ANN*	2.907	0.010	2.865	2.890	2.900	2.908	2.914	2.923	2.942
PSO Gambler Mem1*	2.901	0.025	2.783	2.858	2.884	2.901	2.919	2.942	2.994
Evolved ANN 5 Noise 05*	2.864	0.008	2.830	2.850	2.858	2.865	2.870	2.877	2.891
DBS	2.857	0.009	2.823	2.842	2.851	2.857	2.863	2.872	2.899
Winner12	2.849	0.008	2.820	2.836	2.844	2.850	2.855	2.862	2.874
Fool Me Once	2.844	0.008	2.818	2.830	2.838	2.844	2.850	2.857	2.882
Omega TFT: 3, 8	2.841	0.011	2.800	2.822	2.833	2.841	2.849	2.859	2.882

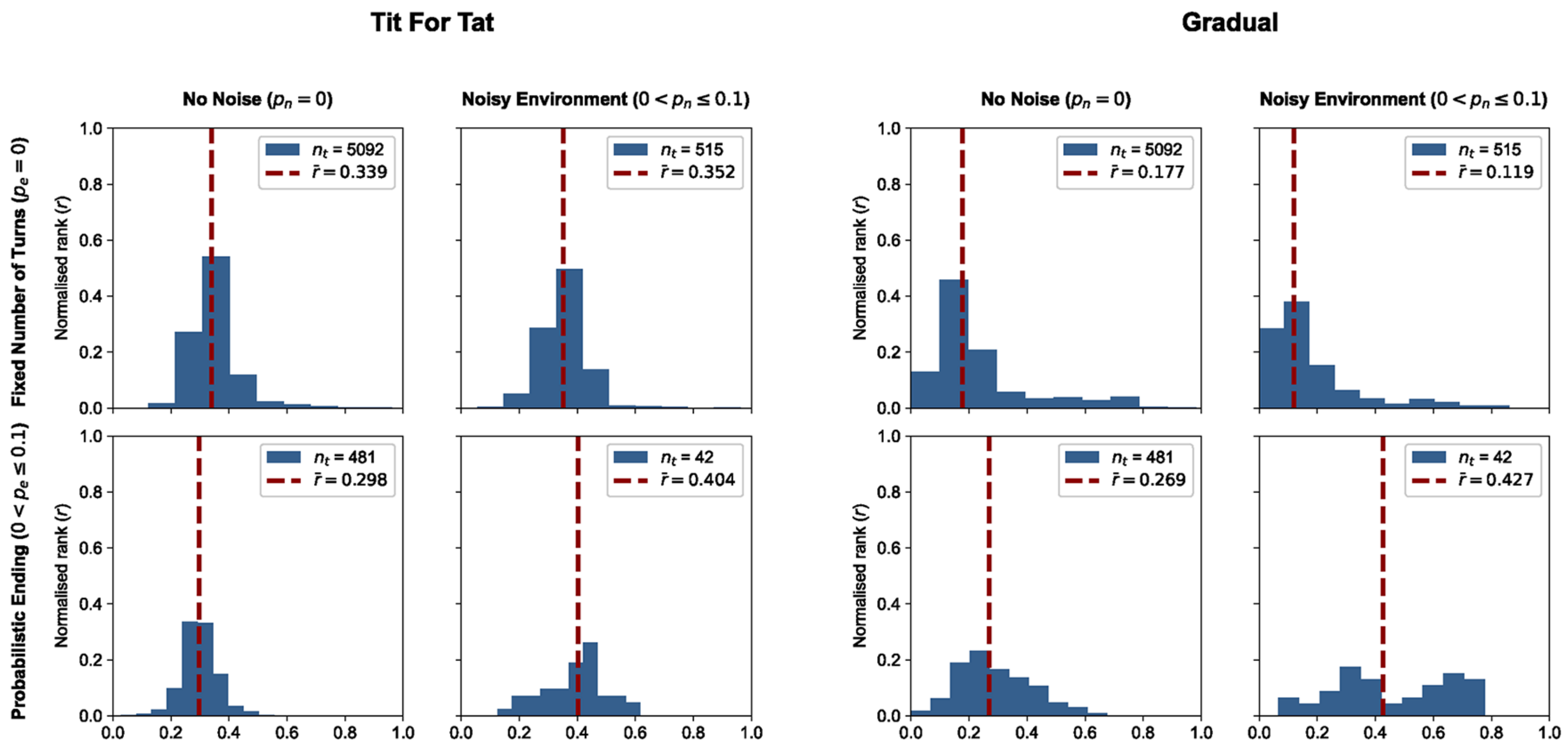
https://doi.org/10.1371/journal.pone.0188046.t002

What does it do?

Explore the properties of winning strategies

Running a large number of tournaments



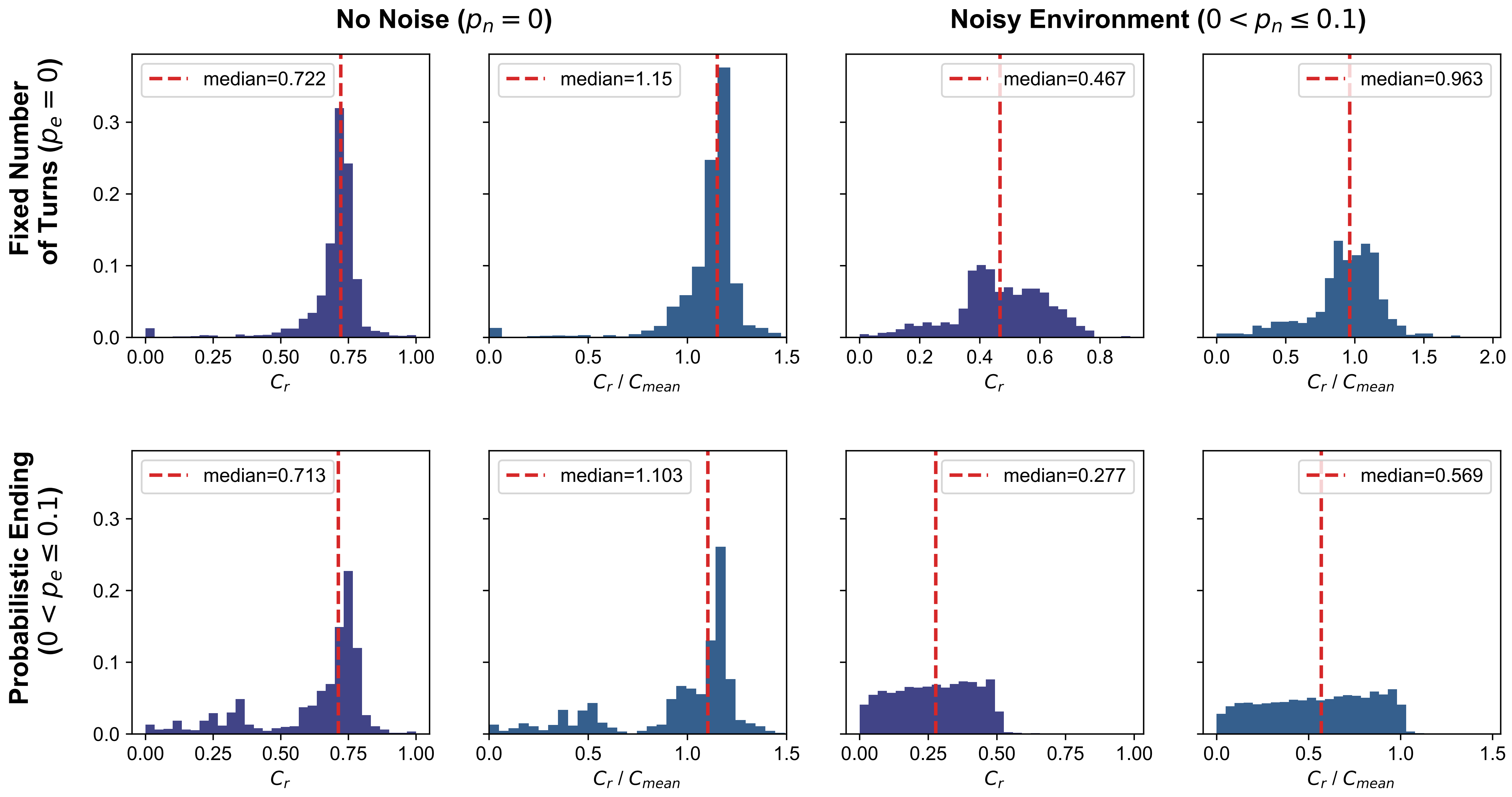


Results

	Standard		Noisy $p_n \leq 0.1$		Probabilistic ending $p_e \leq 0.1$		Noisy probabilistic ending	
	r	median score	r	median score	r	median score	r	median score
CC to C rate	-0.501	0.501	-0.210	0.194	-0.336	0.348	0.087	0.015
CD to C rate	0.226	-0.199	0.337	-0.235	0.458	-0.352	0.609	-0.372
DC to C rate	0.127	-0.100	0.227	-0.111	0.164	-0.105	0.410	-0.203
DD to C rate	0.412	-0.396	0.549	-0.391	0.433	-0.378	0.615	-0.407
C_r	-0.323	0.383	0.298	-0.051	-0.060	0.160	0.595	-0.213
C_{max}	0.000	0.050	-0.000	0.244	-0.000	0.079	-0.000	0.296
C_{min}	0.000	0.085	0.000	-0.070	0.000	0.128	0.000	0.000
C_{median}	0.000	0.209	0.000	0.572	-0.000	0.324	0.000	0.667
C_{mean}	0.000	0.229	-0.000	0.583	-0.000	0.354	-0.000	0.689
C_r / C_{max}	-0.323	0.381	0.307	-0.076	-0.060	0.156	0.608	-0.246
C_{min} / C_r	0.109	-0.080	-0.141	-0.011	0.024	0.029	-0.335	0.092
C_r / C_{median}	-0.330	0.353	0.326	-0.258	-0.065	0.111	0.614	-0.464
C_r / C_{mean}	-0.331	0.357	0.325	-0.228	-0.066	0.114	0.617	-0.431
N	-0.000	-0.009	-0.000	-0.017	-0.000	0.011	0.000	0.139
k	-0.000	-0.002	-0.000	-0.003	-0.000	0.010	-0.000	0.035
n	-0.000	-0.125	-0.000	-0.392	-	-	-	-
p_n	-	-	0.000	-0.244	-	-	0.000	-0.272
p_e	-	-	-	-	0.000	0.257	0.000	0.568
Make use of game	-0.003	-0.022	-0.047	0.014	-0.046	0.022	-0.110	0.057
Make use of length	-0.158	0.124	-0.224	0.139	-0.173	0.128	-0.206	0.115
SSE	0.473	-0.452	0.589	-0.412	0.458	-0.418	0.571	-0.383
stochastic	0.006	-0.024	0.010	-0.007	-0.001	0.001	-0.001	0.002
memory usage	-0.098	0.108	-0.080	0.114	-	-	-	-

https://doi.org/10.1371/journal.pcbi.1012644.t004

Properties



1. Be a little bit envious
2. Be "nice" in non-noisy environments or when game lengths are longer
3. Reciprocate both cooperation and defection appropriately; Be provocable in tournaments with short matches, and generous in tournaments with noise
4. It's ok to be clever
5. Adapt to the environment; Adjust to the mean population cooperation

Analyzing influence

- Can we use methods like random forests, decision trees, or linear regression to identify which strategies most influence the final tournament rankings?

Alternative tournament formats

- What does it mean to be successful?

Summary

Axelrod-Python

https://github.com/Axelrod-Python/Axelrod

https://axelrod.readthedocs.io/en/stable/

https://discord.gg/XCMuZDhQ

Game-theory-summer-school-Kyiv

https://github.com/Nikoleta-v3/Game-theory-summer-school-Kyiv

Contact

Nikoleta-v3

nikoleta.glynatsi@riken.jp

THANK YOU!

Thank you to Oleksii