scientific reports

Check for updates

OPEN

Repeated strategies and negative implementation errors shape induced game types in the iterated Prisoner's Dilemma

Andreas Duerholt

The iterated Prisoner's Dilemma is often studied in terms of strategy performance and long-run cooperation. This paper offers an alternative perspective: given a finite set of strategies, the long-run interaction between strategies is interpreted as an induced symmetric 2 × 2 game, and the distribution of such induced game types is examined under implementation errors. Thirteen strategies are used in large-scale computerized round-robin tournaments. For each noise level, long-run per-round payoffs are estimated by repeated sampling with a geometric stopping rule, and each strategy pair is classified into standard symmetric game types using a payoff-ordering taxonomy. The resulting payoff matrices are then used in a discrete-time replicator dynamic to identify which induced game types characterize the evolved population. In addition, a population-weighted transformation is applied in which payoff entries are weighted by encounter frequencies implied by final population shares, yielding a system-level classification of the effective strategic environment. Whereas the encounter-weighted share of induced Prisoner's Dilemma games is negligible in the final generation under noise-free play, it becomes substantial under negative implementation errors. By contrast, population-weighted classifications concentrate on Concord and Harmony, for the parameter values considered. Bootstrap analysis indicates that the induced game classifications are generally stable under replicate-level resampling.

Keywords Evolutionary dynamics, Game classification, Iterated Prisoner's Dilemma, Repeated games, Tit-for-Tat

The iterated Prisoner's Dilemma (IPD) has been studied extensively, particularly regarding strategy performance and the emergence of long-term cooperation[1–9]. More generally, work on repeated games shows that allowing players to condition on past behavior greatly enlarges the set of sustainable long-run outcomes relative to the one-shot Prisoner's Dilemma[10,11]. From this perspective, the key decision is not a single move to cooperate or defect, but the choice of a strategy (e.g., Tit-for-Tat, Grim Trigger, Always Defect) that prescribes behavior over the course of an ongoing interaction.

Classical computer tournaments showed that simple reciprocal strategies can perform remarkably well. In particular, Tit-for-Tat (TFT) became a benchmark, because it is initially cooperative, retaliates against defection, and returns to cooperation when the opponent does so[12,13]. Subsequent work showed, however, that a strategy's success depends on the surrounding strategy set, the payoff scale and the presence of implementation errors[14–19]. Under noisy conditions, strict reciprocity can lead to retaliation cycles, which motivated strategies designed to restore mutual cooperation after errors, including generous and contrite variants of Tit-for-Tat[13,20]. The strategy Win-Stay-Lose-Shift provides another important benchmark, because it can outperform Tit-for-Tat under suitable conditions and recover cooperation after mutual defection[16,21]. More broadly, work in evolutionary game theory has expanded research in several directions[22]. One strand of this literature examines how mechanisms that support cooperation can alter the effective payoff structure of the Prisoner's Dilemma[23,24]. In evolutionary dynamics with game transitions, the game played in a later period can depend on previous actions and game states[25]. In feedback-evolving games, individual behavior can affect an environmental state, which in turn can change the payoff structure[26,27]. Other modern models study transitions between different games, for example through Markov processes, or variable game interactions in structured populations[28,29]. Together, these studies show that the effective strategic environment need not be fixed.

Institute of Political Science, RWTH Aachen University, Theaterstrasse 35-39, 52072 Aachen, Germany. email: andreas.duerholt@ipw.rwth-aachen.de

The present study is related to this literature but follows a different approach. Here, the stage game remains a Prisoner's Dilemma in every round. What changes is not the underlying one-shot game, but the effective long-run game that emerges when repeated game strategies themselves are treated as the available choices. The question is therefore not whether an external process moves the population from one specified game to another, but which long-run 2×2 game types are induced by canonical IPD strategies, how negative implementation errors alter this induced structure, and how evolutionary selection and population weighting transform the resulting system-level structure.

For example, consider a setting in which only Tit-for-Tat and Always Defect (ALLD) are the strategies available. In the canonical PD with payoffs $T > R > P > S$, TFT can yield high long-run payoffs through mutual cooperation, whereas ALLD against ALLD leads to mutual defection. When TFT plays ALLD, however, it is exploited initially and then retaliates, so the interaction settles into mutual defection. If the continuation probability exceeds the threshold $\delta > \frac{T-R}{T-P}$[1], restricting the strategy set to {ALLD,TFT} induces a long-term interaction that resembles a Stag Hunt (Fig. 1). Similar reasoning applies to other strategy pairs, such as Grim Trigger and ALLD, which can also induce a Stag Hunt[30–33]. Depending on the available strategies and the continuation probability, the repeated interaction can therefore induce a different long-term game.

Despite this observation, it remains unclear which long-run 2×2 game types arise from a broader set of canonical IPD strategies. This study addresses this gap by systematically classifying the effective symmetric 2×2 games induced by pairs of well-known strategies, examining how negative implementation errors alter these classifications, and analyzing how evolutionary weighting changes the prevalence of different induced game types. The study thereby shifts attention from individual strategy performance to the strategic structures generated by the strategy set as a whole.

To address this question, computerized tournaments with 13 canonical strategies are conducted. For each strategy pair, an IPD is simulated and the long-run average payoffs are recorded. These payoffs are used to classify the induced 2×2 game into standard game types, such as Prisoner's Dilemma and Chicken, using the payoff ordering taxonomy proposed by Bruns[34] (see Supplementary Data 1).

The analysis proceeds in four steps. First, round-robin tournaments are simulated in a noise-free environment. Because many strategy pairings sustain mutual cooperation, a substantial fraction of induced interactions is classified as the game "Zero", in which all four payoff entries are equal ($T = R = P = S$).

Second, evolutionary dynamics are introduced, and the population state after 100 generations is used to analyze which induced long-term game types are most prevalent in the evolved population. For each pair of strategies, the induced long-run game is identified from the simulated long-run payoff matrix, and the frequency with which the corresponding matchup occurs is determined by the final population shares. This yields a final-generation distribution of pairwise induced long-run game types.

Third, to examine how this classification changes when mutual cooperation is perturbed, negative implementation errors (ε) are introduced, so that intended cooperation is sometimes implemented as defection. Such errors can alter strategy performance. While previous work has examined the impact of noise on cooperation rates and tournament winners[14–19], less is known about how noise reshapes the distribution of induced game types generated by a heterogeneous strategy population.

Fourth, population-weighted payoff representations are constructed to describe a "system-level structure" of all pairwise classifications. In this transformation, each entry of the long-run payoff matrix is rescaled by the population shares of the two strategies involved. For example, if TFT and ALLD have final population shares of 70% and 30%, respectively, the TFT self-play payoff is weighted by 0.7^2, whereas the payoff from an ALLD-TFT encounter is weighted by 0.3×0.7. The resulting population-weighted payoff entries are then classified with the same payoff-ordering scheme as the unweighted induced games.

Across these analyses, standard errors, and confidence intervals quantify payoff uncertainty, while replicate-level resampling provides bootstrap intervals for game type shares and evaluates classification stability.

The remainder of the paper is organized as follows. The "Methods" section describes the simulation design, the implementation error process, the formal definition of induced games, the replicator-dynamics update rule, the population-weighted classification procedure, and the strategy set considered. The "Results" section presents the noise-free benchmarks, the effects of negative implementation errors, evolved strategy shares, population-weighted classifications, classification stability, and robustness checks. The "Discussion" section discusses the main findings, limitations, and directions for future research.

a

	C	D
C	R, R	S, T
D	T, S	P, P

b

	TFT	ALLD
TFT	R, R	$S + (P - S)\delta, T + (P - T)\delta$
ALLD	$T + (P - T)\delta, S + (P - S)\delta$	P, P

Fig. 1. Stage game and induced long-run game for {TFT,ALLD}. (**a**) One-shot Prisoner's Dilemma with actions cooperation (C) and defection (D), (**b**) Induced long-run payoff matrix when the available choices are Tit-for-Tat (TFT) and Always Defect (ALLD). Payoffs in (**b**) are normalized discounted payoffs under continuation probability δ. The payoff ranking is: $T > R > P > S$. If $\delta > \frac{T-R}{T-P}$, the restricted interaction has the structure of a Stag Hunt.

Methods

This section describes the simulation design, classification procedure and uncertainty analysis used in the present study. It first introduces the stage game and implementation error process, then formalizes the induced game construction and the population-weighted classification procedure. It further describes the evolutionary updating rule based on replicator dynamics, the strategy set considered, and the procedure used to quantify uncertainty.

Tournament design

For the stage game to be a Prisoner's Dilemma, the payoffs are assumed to satisfy $T > R > P > S$ and $2R > T + S$ [1,2,18,35]. These conditions ensure that mutual cooperation is jointly efficient yet individually dominated, and that alternating outcomes cannot yield higher payoffs than sustained cooperation. Across studies, alternative cardinal payoff scales have been used[16,36,37], and results can differ across scales[18]. However, in the Prisoner's Dilemma literature the most common specification is the cardinal scale $(T, R, P, S) = (5, 3, 1, 0)$, with payoffs ranging from 0 to 5, that was used in Axelrod's original computer tournaments[12,13] and has since become a common benchmark in simulations of the iterated Prisoner's Dilemma[38–42].

Here, with a given probability, intended cooperation is implemented as defection, whereas intended defections are always implemented correctly. Such negative noise may be more relevant in political and international contexts than positive or symmetric noise[19,36]. Restricting implementation errors to $C \to D$ keeps the analysis transparent because it isolates a single noise parameter and ensures that unintended deviations affect payoffs only in one direction. Incorporating positive implementation errors ($D \to C$), symmetric implementation errors ($C \leftrightarrow D$), or divergent perception would introduce additional sources of error and make it more difficult to attribute changes in induced games to one specific mechanism. This restriction is discussed as a limitation in the final section.

For each noise level (0–20% in one-percentage-point steps), round-robin tournaments are carried out, and payoffs are estimated by repeated sampling. In Axelrod's first tournament[12], each game consisted of 200 moves, as in later tournaments[36,43]. In the present simulations, the horizon is stochastic rather than fixed. This is advantageous because parity effects are avoided: when two strategies fall into an alternating CD/DC spiral, fixed even horizons could generate ties. To preserve this property while keeping an expected match length of 200 rounds, a continuation probability $\delta = 0.995$ is adopted. In each round, play continues with probability δ and ends with probability $1 - \delta$.

To avoid finite-sample differences in realized match length across strategy pairs, a common random horizon is used within each round-robin replicate and all matches in that replicate are played for exactly that number of rounds. Across replicates, the horizon can vary, but δ is held constant, so that approximately 200 rounds are played on average.

To obtain stable payoff estimates, each unordered pair of distinct strategies is matched 200 times within each replicate. For self-play, each strategy is matched against itself 100 times within each replicate. Because both players use the same strategy, their realized per-round payoffs are averaged and assigned to the corresponding diagonal entry. In total, 100 independent round-robin replicates are conducted per noise level, and the resulting payoff matrices are averaged entry-wise. The final payoff matrix at a given noise level is therefore based on many independent matches and replicated tournaments, so that its entries can be treated as approximations of long-run expected per-round payoffs under the given noise level.

Formally, this simulation procedure defines an expected per-round payoff matrix over the set of repeated game strategies. Let $G^0 = (A, u)$ denote the one-shot Prisoner's Dilemma, where $A = \{C, D\}$ is the set of stage game actions and $u(a, b)$ gives the payoff to a player choosing action $a \in A$ against an opponent choosing action $b \in A$. The payoff parameters are

$$T = u(D, C), R = u(C, C), P = u(D, D), S = u(C, D),$$

with $(T, R, P, S) = (5, 3, 1, 0)$. Let $\Sigma = \{\sigma_1, ..., \sigma_n\}$ denote the finite set of repeated game strategies considered in the tournament. A strategy $\sigma_i \in \Sigma$ maps the observed history of previous play into an intended action in $\{C, D\}$, or into a probability distribution over intended actions. For a given implementation error rate ε, the repeated interaction between two strategies σ_i and σ_j generates a stochastic sequence of realized actions and stage game payoffs. Let L denote the realized length of the interaction, drawn from a geometric stopping rule implied by the continuation probability $\delta = 0.995$. The expected per-round payoff of strategy σ_i against strategy σ_j is defined as

$$\pi_{ij}(\varepsilon) = \mathbb{E}_\varepsilon \left[\frac{1}{L} \sum_{t=1}^{L} u\left(a_i^t, a_j^t\right) \right],$$

where a_i^t and a_j^t are the realized actions of strategies σ_i and σ_j in round t. The expectation ($\mathbb{E}_\varepsilon$) is taken over the stochastic interaction process at implementation error rate ε, including random match lengths, and any stochastic elements in the strategies themselves. In practice, $\pi_{ij}(\varepsilon)$ is estimated by repeated simulations and tournament replicates, as described above. Collecting these estimates for all strategy pairs yields the long-run payoff matrix

$$\Pi(\varepsilon) = \left[\pi_{ij}(\varepsilon)\right]_{i,j=1}^{n}.$$

For any unordered pair of distinct strategies $\{\sigma_i, \sigma_j\}$, the corresponding induced game is defined as the symmetric two-strategy normal-form game in which the available choices are not the stage game actions C and D, but the repeated strategies σ_i and σ_j. This induced game is denoted by $G_{ij}^{ind}(\varepsilon)$, where the superscript "ind" refers to "induced". To write a payoff matrix, let σ_i be the first option and σ_j the second option. The induced game is represented by the row-player payoff matrix

$$G_{ij}^{ind}(\varepsilon) = \begin{pmatrix} \pi_{ii}(\varepsilon) & \pi_{ij}(\varepsilon) \\ \pi_{ji}(\varepsilon) & \pi_{jj}(\varepsilon) \end{pmatrix},$$

where

$$\mathrm{R}'_{\mathrm{ij}} = \pi_{\mathrm{ii}}(\varepsilon),\ \mathrm{S}'_{\mathrm{ij}} = \pi_{\mathrm{ij}}(\varepsilon),\ \mathrm{T}'_{\mathrm{ij}} = \pi_{\mathrm{ji}}(\varepsilon),\ \mathrm{P}'_{\mathrm{ij}} = \pi_{\mathrm{jj}}(\varepsilon).$$

Equivalently, using the standard notation for symmetric 2×2 games, this matrix can be written as

$$G_{ij}^{ind}(\varepsilon) = \begin{pmatrix} \mathrm{R}'_{\mathrm{ij}} & \mathrm{S}'_{\mathrm{ij}} \\ \mathrm{T}'_{\mathrm{ij}} & \mathrm{P}'_{\mathrm{ij}} \end{pmatrix}.$$

Thus, the term "induced game" refers to the strategic structure generated by the long-run interaction of the two repeated game strategies when these strategies themselves are treated as available choices. The stage game remains a Prisoner's Dilemma in every round, whereas the induced game describes the payoff ordering generated by the repeated strategies.

Based on these induced payoff entries $(\mathrm{R}'_{\mathrm{ij}}, \mathrm{S}'_{\mathrm{ij}}, \mathrm{T}'_{\mathrm{ij}}, \mathrm{P}'_{\mathrm{ij}})$, each induced game is classified as a standard symmetric 2×2 game type, such as Stag Hunt, Chicken or Prisoner's Dilemma. Since the pair $\{\sigma_i, \sigma_j\}$ is unordered, the classification is invariant to relabeling the two strategies. Relabeling σ_i and σ_j corresponds to swapping both rows and columns of the induced payoff matrix. For example, both

$$\mathrm{T}'_{\mathrm{ij}} > \mathrm{R}'_{\mathrm{ij}} > \mathrm{P}'_{\mathrm{ij}} > \mathrm{S}'_{\mathrm{ij}}$$

and the relabeled ordering

$$\mathrm{S}'_{\mathrm{ji}} > \mathrm{P}'_{\mathrm{ji}} > \mathrm{R}'_{\mathrm{ji}} > \mathrm{T}'_{\mathrm{ji}}$$

are classified as Prisoner's Dilemma. The resulting classification identifies the induced long-run game type for each strategy pair, using a tie tolerance of 10^{-10}.

Let $x_i(\varepsilon)$ denote the final population share of strategy σ_i at implementation error rate ε, with

$$\sum_{i=1}^{n} x_i(\varepsilon) = 1.$$

For each unordered pair of distinct strategies $\{\sigma_i, \sigma_j\}$, the encounter weight is $w_{ij}(\varepsilon) = x_i(\varepsilon)\, x_j(\varepsilon)$. Thus, induced game types generated by common strategy pairs contribute more strongly to the reported final-generation frequencies than induced game types generated by rare strategy pairs.

In addition, a population-weighted payoff transformation is constructed. For this transformation, each entry of the expected per-round payoff matrix is multiplied by the product of the final population shares of the two strategies involved:

$$\pi_{ij}^{*}(\varepsilon) = x_i(\varepsilon)\, x_j(\varepsilon)\, \pi_{ij}(\varepsilon).$$

This yields a population-weighted payoff matrix

$$\Pi^{*}(\varepsilon) = \left[\pi_{ij}^{*}(\varepsilon)\right]_{i,j=1}^{n}.$$

For each unordered pair $\{\sigma_i, \sigma_j\}$, the corresponding population-weighted payoff representation is

$$G_{ij}^{*}(\varepsilon) = \begin{pmatrix} \pi_{ii}^{*}(\varepsilon) & \pi_{ij}^{*}(\varepsilon) \\ \pi_{ji}^{*}(\varepsilon) & \pi_{jj}^{*}(\varepsilon) \end{pmatrix}.$$

Equivalently, the population-weighted payoff entries are

$$\mathrm{R}_{\mathrm{ij}}^{*} = \pi_{ii}^{*}(\varepsilon),\ \mathrm{S}_{\mathrm{ij}}^{*} = \pi_{ij}^{*}(\varepsilon),\ \mathrm{T}_{\mathrm{ij}}^{*} = \pi_{ji}^{*}(\varepsilon),\ \mathrm{P}_{\mathrm{ij}}^{*} = \pi_{jj}^{*}(\varepsilon).$$

The population-weighted payoff representation is classified using the same payoff-ordering scheme as the pairwise induced game. This transformation defines a system-level payoff representation whose entries combine long-run payoffs with population encounter frequencies; its classification describes the ordering of these weighted entries rather than the strategic incentives of a conventional two-player game. Primes denote pairwise induced game payoffs, whereas asterisks denote population-weighted payoff entries. The frequencies of the population-

weighted classifications are summarized using the same normalized encounter-weighted aggregation procedure as above.

The long-run payoff matrix $\Pi(\varepsilon)$ is also used as input for an evolutionary process. Population updating is implemented using a discrete-time replicator dynamic, a standard deterministic model of frequency-dependent selection in which strategies with fitness above the population average increase in frequency. The rule is used here as a transparent baseline for translating payoff differences among strategies into changes in population composition. Closely related discrete-time replicator-mutator dynamics continue to be used in recent evolutionary game applications, including a study of social learning[44]. Let $x_i(t,\varepsilon)$ denote the population share of strategy σ_i in generation t at implementation error rate ε, with $\sum_{i=1}^{n} x_i(t,\varepsilon) = 1$. The expected payoff, or fitness, of strategy σ_i is

$$f_i(t,\varepsilon) = \sum_{j=1}^{n} x_j(t,\varepsilon)\,\pi_{ij}(\varepsilon).$$

The mean population payoff is

$$\overline{f}(t,\varepsilon) = \sum_{i=1}^{n} x_i(t,\varepsilon)\,f_i(t,\varepsilon).$$

Population shares are updated according to the discrete-time replicator equation

$$x_i(t+1,\varepsilon) = x_i(t,\varepsilon)\,\frac{f_i(t,\varepsilon)}{\overline{f}(t,\varepsilon)}.$$

The evolutionary process is run for 100 generations at each noise level. To examine whether the results depend on this finite evolutionary horizon, an additional robustness check is conducted in which the process is run for 500 generations.

The simulations are run for noise levels from 0 to 20%. This range covers low to relatively high implementation error rates while keeping the computational design tractable. To visualize how the shares of induced game types vary with implementation errors, LOWESS (locally weighted scatterplot smoothing, $frac = 0.25$) is applied, using the noise level as the independent variable and the share of the induced game type as the dependent variable.

Uncertainty quantification and bootstrap analysis

To quantify uncertainty in the estimated payoffs and induced game classifications, the complete payoff matrix obtained in each of the 100 independent round-robin replicates is retained. For each payoff entry, the mean, standard deviation, and standard error are calculated across replicates. A 95% confidence interval is constructed using the Student's t-distribution with 99 degrees of freedom. Because the classification of an induced game depends on the relations among T, R, P, and S, uncertainty is also calculated for the six pairwise contrasts $T-R$, $T-P$, $T-S$, $R-P$, $R-S$, and $P-S$. These contrasts indicate whether the payoff differences that determine a classification are clearly separated from zero or whether the induced game lies close to a payoff-ordering boundary.

The game classification itself is categorical and therefore does not have a conventional numerical standard error or confidence interval. Classification uncertainty is instead summarized by bootstrap support and classification probabilities. For each noise level, 5000 bootstrap samples are generated by drawing 100 complete round-robin matrices with replacement from the 100 original replicate matrices. Complete matrices are resampled because all pairings within a replicate share the same randomly drawn horizon. This procedure therefore preserves the dependence structure generated by the common random horizon.

Within each bootstrap sample, the selected payoff matrices are averaged entry-wise, the replicator dynamic is rerun for 100 generations, and the final population shares are recalculated. The pairwise induced games and the population-weighted representations are then reconstructed and classified using the same payoff-ordering scheme and numerical tolerance as in the main analysis. Bootstrap support is defined as the proportion of bootstrap samples in which the classification obtained from the averaged baseline payoff matrix is reproduced. For the encounter-weighted shares of each game type, bootstrap standard errors are calculated as the standard deviation of the bootstrap estimates. Both normal-approximation 95% confidence intervals and percentile bootstrap intervals, defined by the 2.5th and 97.5th percentiles, are calculated. Supplementary Data 2 reports the distribution of bootstrap support, all cases in which the classification obtained from the averaged payoff matrix differs from the most frequent bootstrap classification, and, for each noise level, the estimated share, bootstrap standard error, normal-approximation 95% confidence interval, and percentile bootstrap interval of the most prevalent game type in the unweighted and population-weighted analyses.

Selection of strategies

A central design choice concerns which strategies to include in the tournament, balancing computational tractability against the aim of presenting a broad range of strategies. In total, 13 strategies are included: Always Defect, Alternator, Adaptive Pavlov, Cooperate After Mutual Punishment, Contrite Tit-for-Tat, Don't Look Back in Anger, Gradual, Grim Trigger, Generous Tit-for-Tat, JOSS, Suspicious Tit-for-Tat, Tit-for-Two-Tats, and Win-Stay-Lose-Shift. The selection is guided both by prior performance in computer tournaments and the

plausibility of the strategies as behavioral models. For instance, Always Defect (ALLD) has often performed poorly in computer tournaments, but it remains a natural benchmark because defection is strictly dominant in the one-shot Prisoner's Dilemma. In other cases, the specific implementation to be adopted is less obvious because multiple variants of some canonical strategies exist, such as Tit-for-Tat. For this reason, several widely used variants of TFT are included here.

Contrite Tit-for-Tat (CTFT) reduces to standard TFT in error-free interactions, while incorporating a contrition mechanism that helps sustain cooperation under implementation errors. When CTFT defects unintentionally, it does not answer the opponent's next retaliatory defection with further punishment, allowing cooperation to be restored[45]. Generous Tit-for-Tat (GTFT) is a probabilistic variant of Tit-for-Tat in which observed defection is answered by cooperation with a fixed probability of 10%. By introducing leniency, the strategy can break mutual defection cycles[14]. The strategy JOSS is analogous but introduces occasional opportunistic defections instead of forgiveness: after an opponent's cooperative move, JOSS defects with probability 10%[46].

Tit-for-Two-Tats (TF2T) responds with defection only when the opponent has defected in two successive rounds[46,47]. TF2T is generally less vulnerable to negative noise but it can be exploited more easily by strategies such as Alternator (ALTR), which tries to switch between cooperation and defection[36]. Don't Look Back in Anger (DLBA) and Cooperate After Mutual Punishment (CAMP) are also considered because they have been found to perform well under noisy conditions[36]. CAMP begins with two cooperative moves and switches to defection only when the previous round yielded the sucker's payoff. After mutual defection, it returns to cooperation in the hope that the opponent follows. DLBA combines features of TF2T and CAMP: it starts by cooperating in the first two rounds, and retaliation is triggered after receiving the sucker's payoff, unless the temptation payoff was received in the preceding round. It also defects when receiving the punishment payoff, but to avoid getting trapped in continuous punishment phases, DLBA resumes cooperation if the punishment payoff has occurred in two successive rounds[36].

The strategy Win-Stay-Lose-Shift (WSLS), sometimes termed Pavlov[16,48], can outperform TFT under suitable conditions[21]. The strategy begins with cooperation and then maintains its current choice when the previous round produced either the reward or the temptation payoff, whereas it changes action after receiving the sucker's payoff or the punishment payoff[49]. Both WSLS and CAMP therefore return to cooperation after mutual defection, but their responses differ after receiving the temptation payoff.

Because CAMP, DLBA, and TF2T begin by cooperating in the first two rounds, Suspicious Tit-for-Tat (STFT) is also included as a contrasting strategy. STFT defects in the very first round and afterwards imitates the opponent's previous move[47,50]. This can secure at least a tie in many matchups and can permit exploitation of overly cooperative opponents at the beginning. Conversely, STFT can be more vulnerable to defection spirals in the presence of noise.

In the tournament of Beaufils, Delahaye and Mathieu[51], the strategy Gradual emerged as the winner. Gradual cooperates initially, but after an observed defection, a punishment phase is triggered in which defection is played a number of times equal to the opponent's cumulative defections so far, after which two cooperative "cool-down" rounds are played, before any further punishment phase begins. For example, after the opponent's fourth defection, Gradual defects four times in a row and then cooperates for two rounds. In this way, Gradual promotes cooperation through increasingly long but finite punishment phases.

Twenty years after Axelrod's round-robin tournaments, additional canonical strategies were evaluated, including Adaptive Pavlov (AP)[40]. In its modern implementation, which is used here, AP behaves like TFT for the first six moves and then switches among TF2T, TFT or ALLD modes depending on the observed frequency of defections. This mode is re-evaluated every six rounds[52]. Finally, Grim Trigger (GT), sometimes called "spiteful"[47], is also included because it remains one of the best-known strategies. Grim maintains cooperation until the opponent defects for the first time, after which it defects permanently. The complete baseline simulation results are provided in Supplementary Data 3.

Results

This section reports how the distribution of induced game types changes across noise levels and population states. It presents the noise-free benchmark, the effects of negative noise, the strategies that become prevalent under replicator dynamics, the population-weighted classifications, classification uncertainty, and robustness checks.

Noise-free play

In the noise-free round-robin benchmark, the Prisoner's Dilemma is induced in only five of the 78 pairwise long-run classifications. The only exceptions occur when ALLD is paired with ALTR, CAMP or WSLS, or when Alternator is combined with AP or TF2T (counting each unordered pair only once). After 100 generations, these Prisoner's Dilemma classifications become negligible because ALLD or ALTR are very rare in the evolved population.

The game "Zero" is the most common classification, occurring in 37 of the 78 pairings. In most of these cases, both strategies cooperate initially and maintain cooperation along histories of mutual cooperation, yielding equal payoffs in self-play and cross-play. Their differing retaliation and forgiveness mechanisms remain inactive along these noise-free histories but can generate different responses following implementation errors, potentially breaking the payoff equality. However, Zero can also arise through persistent mutual defection, as in ALLD-STFT.

Concord ($R' > T' > S' > P'$), is the second-most common classification, occurring in 14 pairings, especially those involving ALTR, JOSS, or STFT. For example, in pairings of CTFT with ALTR, JOSS, or STFT, choosing CTFT against itself yields the highest payoff. When CTFT is paired with ALTR or STFT, play can enter alternating

cooperation-defection patterns. So, when CTFT is paired with JOSS, repeated opportunistic defections by JOSS can lead to persistent punishment payoffs.

All eight Stag Hunt ($R' > T' > P' > S'$) classifications occur when ALLD is one of the two available strategies. In these cases, self-play by the cooperative strategy yields the highest payoff but choosing it against ALLD yields the lowest payoff. Thus, if the opponent chooses ALLD, responding with ALLD is optimal within the restricted strategy set.

By contrast, long-run Chicken Games ($T' > R' > S' > P'$) are induced in eight pairings. Seven involve STFT or JOSS, the only exception is ALTR-DLBA. STFT's initial defection can create a payoff asymmetry against CAMP, DLBA, Gradual, and TF2T, which begin cooperatively. JOSS can generate a similar "Chicken advantage" against forgiving strategies (e.g., DLBA) or strategies that tolerate single defections (e.g., TF2T).

Other induced game types include Assurance ($R' > P' > T' > S'$) when Grim Trigger and JOSS are the strategies available, Peace ($R' > S' > P' > T'$) when Alternator is paired with Gradual or Grim Trigger, and Battle ($T' > S' > R' > P'$) when JOSS and STFT are paired with each other or with ALTR. In one pairing, a "Tie" is obtained because STFT's first round advantage is offset by Grim's permanent punishment from the second round onward. For a clearer overview, Fig. 2 shows all induced classifications at 0% implementation error.

After replicator updating under noise-free conditions, the game Zero is the most prevalent classification in the encounter-weighted distribution because ALLD, ALTR, JOSS, and STFT have low fitness and become rare. Most encounters therefore occur among "nice" reciprocators that sustain mutual cooperation. The population-weighted noise-free outcome is instead classified as Tie, which will be discussed in "Population-weighted structures" section.

Long-run games under implementation errors

Once implementation errors are introduced, the game "Zero" occurs less often because cooperative strategies occasionally defect, even though defection is unintended. Negative noise therefore breaks endless cooperation and reshapes the distribution of induced long-run game types without changing the stage-game payoffs.

Between 1 and 8% noise, the final-generation share of Concord generally increases to about 30%, before declining to around 10% at higher noise rates. Chicken rises to almost 40% at $\varepsilon = 1\%$, remains above one third through $\varepsilon = 8\%$, and then declines to almost zero at $\varepsilon = 20\%$.

The share of Prisoner's Dilemma also rises initially, exceeding 25% at $\varepsilon = 1\%$. It generally remains between approximately 20% and 40%, but falls below 10% at 11%, 13%, and 20% noise. Conversely, Deadlock appears only at noise rates of 11%, 13%, and 20%, with shares ranging from approximately 20% to more than 35%. This indicates a substitution pattern between Prisoner's Dilemma and Deadlock. Figure 3 shows the final-generation shares of Chicken, Concord, Deadlock, and Prisoner's Dilemma.

The Stag Hunt remains rare in the final-generation distribution, except at noise rates of 1% and 9%, where its share lies between 15 and 25%. Coordination ($R' > P' > S' > T'$) and Peace ($R' > S' > P' > T'$) occur at higher noise rates ($\varepsilon > 10\%$). As with the substitution pattern between Deadlock and Prisoner's Dilemma, Peace and Coordination tend to appear at different noise rates, with shares between 25 and 45%. Several other game types, such as Compromise ($S' > R' > T' > P'$), Harmony ($R' > S' > T' > P'$), and Hero ($S' > T' > R' > P'$), appear only sporadically.

	ALLD	ALTR	AP	CAMP	CTFT	DLBA	Grad	GT	GTFT	Joss	STFT	TF2T	WSLS
ALLD		PD	SH	PD	SH	SH	SH	SH	SH	SH	Z	SH	PD
ALTR	PD		PD	CC	CC	CG	Pe	Pe	CC	Ba	Ba	PD	CC
AP	SH	PD		Z	Z	Z	Z	Z	Z	CC	CC	Z	Z
CAMP	PD	CC	Z		Z	Z	Z	Z	Z	CC	CG	Z	Z
CTFT	SH	CC	Z	Z		Z	Z	Z	Z	CC	CC	Z	Z
DLBA	SH	CG	Z	Z	Z		Z	Z	Z	CG	CG	Z	Z
Grad	SH	Pe	Z	Z	Z	Z		Z	Z	CC	CG	Z	Z
GT	SH	Pe	Z	Z	Z	Z	Z		Z	AS	Tie	Z	Z
GTFT	SH	CC	Z	Z	Z	Z	Z	Z		CC	CC	Z	Z
Joss	SH	Ba	CC	CC	CC	CG	CC	AS	CC		Ba	CG	CC
STFT	Z	Ba	CC	CG	CC	CG	CG	Tie	CC	Ba		CG	CC
TF2T	SH	PD	Z	Z	Z	Z	Z	Z	Z	CG	CG		Z
WSLS	PD	CC	Z	Z	Z	Z	Z	Z	Z	CC	CC	Z	

Fig. 2. Induced game types at 0% implementation error. Game types are classified from the unweighted long-run payoff matrix estimated by round-robin simulations with expected match length 200 rounds, geometric stopping rule (δ=0.995), and 100 round-robin replicates. Within each replicate, 100 self-play matches are simulated for each diagonal entry, with both players' payoffs assigned to the corresponding self-play payoff, and 200 matches are simulated for each cross-strategy pair. Off-diagonal cells show the induced game type for each strategy pair; diagonal entries are omitted, because self-play does not define a distinct two-strategy induced game. Abbreviations: AS = Assurance; Ba = Battle; CC = Concord; CG = Chicken; PD = Prisoner's Dilemma; Pe = Peace; SH = Stag Hunt; Tie = payoff-ordering tie; Z = Zero.

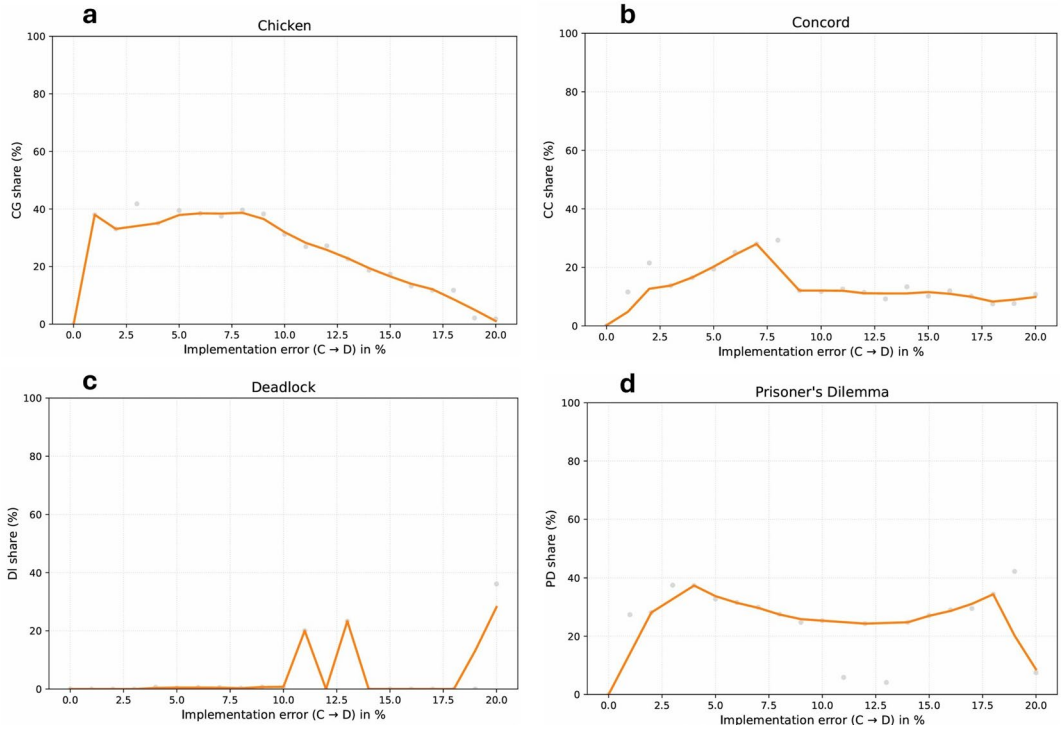


Fig. 3. Selected pairwise induced game types in the final generation. Shares of (**a**) Chicken, (**b**) Concord, (**c**) Deadlock, and (**d**) Prisoner's Dilemma in the baseline 13-strategy tournament after 100 generations. Payoffs are estimated from 100 round-robin replicates per noise level, with expected match length 200 rounds under a geometric stopping rule (δ=0.995). Within each replicate, 100 self-play matches are simulated for each diagonal entry, with both players' payoffs assigned to the corresponding self-play payoff, and 200 matches are simulated for each cross-strategy pair. The x-axis shows the implementation error rate (CD); the y-axis shows the final-generation share of the corresponding induced game type. Gray points show simulated values, and orange lines show LOWESS-smoothed trends ($frac = 0.25$). CG = Chicken Game; CC = Concord; Dl = Deadlock; PD = Prisoner's Dilemma.

By contrast, Assurance ($R' > P' > T' > S'$), Battle ($T' > S' > R' > P'$), Zero and other "Ties" almost never emerge in the final generation distribution of induced game types. Given the large number of replicates, the presence of noise, and the small numerical tie tolerance (10^{-10}), payoff ordering ties are unlikely to occur. The contrast is nevertheless notable because Assurance and Battle occur in several pairings in the noise-free round-robin benchmark, whereas Compromise, Coordination, Harmony, and Hero do not. The evolutionary process and the implementation error rate therefore change which induced game types are represented in the final-generation distribution, raising the question of why some classifications persist while others disappear.

Strategies as the main drivers

Once the implementation error rate exceeds 4%, three strategies become most prevalent under replicator dynamics: CAMP, CTFT, and DLBA. At very low noise rates, TF2T and Gradual also retain substantial population shares. The restriction to negative implementation errors helps explain the strong performance of CTFT in these simulations. In other simulations, TF2T often emerges as more successful, because CTFT is more vulnerable when perception can diverge[36]. Under the present assumptions, in which only implementation errors that turn intended cooperation into defection are allowed, CTFT's advantage is preserved[20]. To clarify the strategic structure implied by this prevalence pattern, Fig. 4 reports the long-run payoffs among CAMP, CTFT, and DLBA at implementation error rates of 12% and 13%.

For the CAMP-CTFT pair, the induced game is Coordination at both 12% and 13% noise, whereas the CAMP-DLBA pair is classified as Chicken. By contrast, the CTFT-DLBA pair is classified as Prisoner's Dilemma at $\varepsilon = 12\%$ and Deadlock at $\varepsilon = 13\%$. Because CTFT and DLBA are highly prevalent, this pairing contributes strongly to the corresponding final-generation shares of Prisoner's Dilemma and Deadlock.

The switch between Prisoner's Dilemma and Deadlock is driven by very small differences between the CTFT and DLBA self-play payoffs (Table 1). With DLBA listed as the first strategy, R' denotes DLBA self-play and P' denotes CTFT self-play. When $R' > P'$, the CTFT-DLBA pairing is classified as Prisoner's Dilemma. When $P' > R'$, it is classified as Deadlock[53]. In these cases, the CTFT-DLBA payoff ordering shifts to Deadlock, where the dominant-strategy outcome does not lead to a Pareto-inferior outcome, because $T' > P' > R' > S'$. The 95% confidence intervals for $R' - P'$ include zero at all selected noise rates, indicating that the classification is sensitive to replicate-level variation. Bootstrap support nevertheless exceeds 80% at all selected noise rates except $\varepsilon = 20\%$.

a

	CAMP	CTFT	DLBA
CAMP	2.552 , 2.552	2.538 , 2.551	2.876 , 2.559
CTFT	2.551 , 2.538	2.764 , 2.764	2.783 , 2.735
DLBA	2.559 , 2.876	2.735 , 2.783	2.766 , 2.766

b

	CAMP	CTFT	DLBA
CAMP	2.524 , 2.524	2.513 , 2.523	2.862 , 2.564
CTFT	2.523 , 2.513	2.741 , 2.741	2.758 , 2.712
DLBA	2.564 , 2.862	2.712 , 2.758	2.740 , 2.740

Fig. 4. Payoff matrices for prevalent strategies at 12% and 13% implementation error. Mean normalized per-round payoffs among CAMP, CTFT, and DLBA at (**a**) 12% and (**b**) 13% negative implementation error (*CD*), estimated from 100 round-robin replicates with expected match length 200 rounds ($\delta = 0.995$). Within each replicate, 100 self-play matches are simulated for each diagonal entry, with both players' payoffs assigned to the corresponding self-play payoff, and 200 matches are simulated for each cross-strategy pair. Each cell reports the payoff pair for the row strategy and the column strategy, respectively. Entries are rounded to three decimals for display; classifications use unrounded values with numerical tolerance 10^{-10}.

Noise rate	R′ (DLBA-DLBA)	P′ (CTFT-CTFT)	R′ − P′	Game type	SE	95% *t* CI	Bootstrap game type support (%)
10%	2.804	2.803	0.0011	PD	0.0011	[−0.0011, 0.0033]	84.1
11%	2.784	2.785	−0.0011	Dl	0.0012	[−0.0034, 0.0013]	82.1
12%	2.766	2.764	0.0015	PD	0.0011	[−0.0007, 0.0037]	91.7
13%	2.740	2.741	−0.0008	Dl	0.0008	[−0.0024, 0.0007]	85.6
14%	2.727	2.724	0.0032	PD	0.0019	[−0.0006, 0.0070]	98.2
19%	2.623	2.621	0.0013	PD	0.0015	[−0.0016, 0.0042]	80.1
20%	2.607	2.607	-0.0002	Dl	0.0017	[−0.0034, 0.0031]	55.6

Table 1. Payoff-ordering sensitivity of the DLBA-CTFT pairing. R' and P' are the mean normalized per-round self-play payoffs for DLBA and CTFT, respectively, with DLBA listed as the first strategy. The standard error and 95% *t* confidence interval refer to the contrast $R' - P'$ across the 100 independent round-robin replicates. Bootstrap support is the proportion of the 5000 replicate-level bootstrap samples that reproduce the reported classification. PD = Prisoner's Dilemma, Dl = Deadlock.

The Deadlock structure captures that one "cooperative" strategy (CTFT) systematically outperforms another "cooperative" strategy (DLBA) in the long-run, because in the induced 2×2 representation choosing CTFT strictly "dominates" choosing DLBA. The labels "cooperation" and "defection" in the 2×2 representation should therefore be understood as labels for choosing one repeated game strategy rather than the other, not as stage game actions. Swapping the labels of the two strategies yields an equivalent classification, because the payoff matrix is classified up to relabeling of rows and columns.

Population-weighted structures

Finally, the population-weighted representations are considered. Only a small subset of classifications is observed after population weighting. At $\varepsilon = 0\%$, the population-weighted distribution is classified as a "Tie" rather than Zero. Zero requires exact equality across all long-run payoffs ($R^* = T^* = S^* = P^*$). Such equality can occur if the pairwise payoffs are equal and the corresponding entries receive equal population weights, or if differences in the payoffs are exactly offset by differences in population weights. Under noise-free play, cooperative strategies earn essentially identical per-round payoffs, whereas "harsh" strategies decline and contribute little to aggregate interaction frequencies. At the same time, small differences in population shares persist among the cooperative strategies. Most notably, Gradual attains the largest share, and some strategies perform better than others in the early generations. These small differences break the equality required for Zero and instead yield the payoff ordering ($R^* > T^* = S^* > P^*$).

Once noise is introduced, the population-weighted classifications become concentrated primarily on Concord and Harmony (Fig. 5). In both classifications, R^* yields the highest payoff and P^* the lowest payoff. This raises two questions: why population-weighting favors Concord and Harmony, and why the ordering shifts from Concord to Harmony as the noise rate increases.

Although CTFT and DLBA induce a long-run Prisoner's Dilemma or Deadlock in the unweighted classification (if $\varepsilon \geq 1\%$), their system-level contribution depends on their final population shares. The same mechanism applies when the unweighted induced game is Chicken (CAMP-DLBA) or Coordination (CTFT-CAMP). In the population-weighted transformation, self-play payoffs are weighted by x_i^2 or x_j^2, whereas cross-strategy payoffs are weighted by $x_i x_j$. When population shares are sufficiently unequal, the off-diagonal entries therefore become less influential relative to the self-play payoff of the more prevalent strategy. Depending on which strategy, or strategy cluster, is prevalent, this can yield a population-weighted structure of either Concord ($R^*>T^*>S^*>P^*$) or Harmony ($P^*>T^*>S^*>R^*$), equivalently ($R^*>S^*>T^*>P^*$) after relabeling.

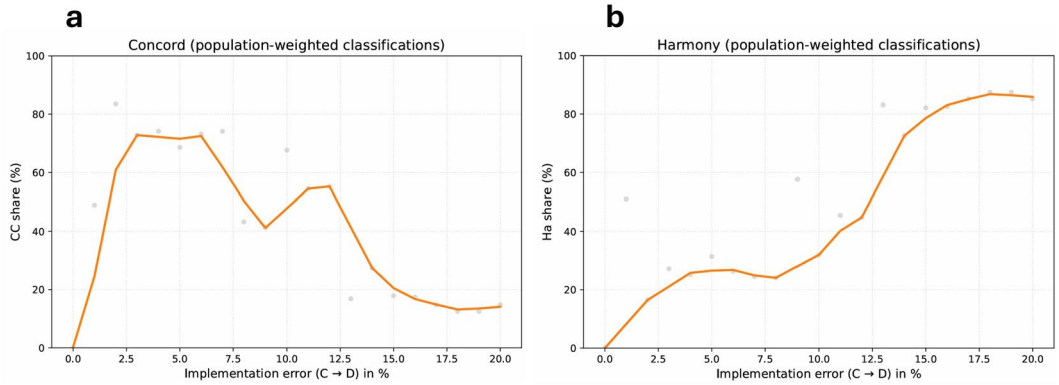


Fig. 5. Population-weighted classifications in the final generation. Population-weighted classifications after 100 generations in the baseline 13-strategy tournament: (**a**) Concord and (**b**) Harmony. Estimates are based on 100 independent round-robin replicates per noise level, with an expected match length of 200 rounds (δ=0.995) . Within each replicate, 100 self-play matches are simulated for each diagonal entry, with both players' payoffs assigned to the corresponding self-play payoff, and 200 matches are simulated for each cross-strategy pair. The x-axis shows the implementation error rate (CD); the y-axis shows the final-generation share of the corresponding population-weighted payoff classification. Gray points show simulated values, and orange lines show LOWESS-smoothed trends ($frac = 0.25$). Abbreviations: CC = Concord; Ha = Harmony.

Therefore, the aggregate classification gives greater weight to outcomes that arise within the most prevalent strategy cluster. Accordingly, an ordering in the unweighted induced game does not automatically carry over to the population-weighted classification (e.g., $R' > P'$ does not necessarily imply $R^* > P^*$). Thus, even though the stage game remains a Prisoner's Dilemma, and even though some pairwise induced games are classified as Prisoner's Dilemma, Deadlock, Chicken, or Coordination, the population-weighted distribution can be concentrated on Concord and Harmony. The remaining question is therefore which strategy, or strategy cluster, becomes prevalent enough to shape this system-level structure.

The transition from Concord to Harmony is mainly driven by the increasing prevalence of CTFT under replicator dynamics. When CTFT becomes a large share of the population, the population-weighted analysis strongly emphasizes CTFT–CTFT interactions relative to interactions involving rare strategies. CTFT and CAMP induce the game Coordination ($R' > P' > S' > T'$), which is the most frequently induced game at implementation error rates of 12% and 13%. As CTFT becomes increasingly common, CTFT-CTFT interactions receive greater weight in the system-level payoff representation, shifting the population-weighted classification from Concord toward Harmony (Fig. 5) even though the relevant CTFT-CAMP pairwise induced game remains Coordination (Fig. 4). Figure 6 illustrates the population-level shift by comparing the evolutionary trajectories at 12% and 13% implementation error.

At noise rates between 12 and 13%, CAMP is overtaken by CTFT (Fig. 6), and the population-weighted classification shifts accordingly. A small core of high-frequency strategies generates most weighted matchups, whereas interactions involving low-frequency strategies contribute little to the population-weighted payoff ordering. Since strategies can overtake one another during the evolutionary process, the population-weighted distribution also depends on the generation at which it is measured. Early transient populations can therefore differ from later, more concentrated populations, which can change the population-weighted classification.

Uncertainty and classification stability

Across the 1,638 strategy-pair and implementation error combinations, the classification obtained from the averaged payoff matrix received at least 95% bootstrap support in 89.6% of the unweighted classifications and 81.6% of the population-weighted classifications. Bootstrap exceeded 80% in 93.7% and 88.2% of cases, respectively. The classification obtained from the averaged payoff matrix differed from the most frequent bootstrap classification in only 6 unweighted and 20 population-weighted cases, corresponding to 26 of the 3,276 classifications across both analyses (0.8%). Thirteen of these 26 cases involved one of the stochastic strategies GTFT and JOSS. Thus, most induced game classifications are stable under replicate-level sampling.

Uncertainty is greater when payoff differences are close to zero, especially for pairings that switch between Prisoner's Dilemma and Deadlock or Peace and Coordination. Nevertheless, the principal aggregate patterns remain. Supplementary Data 2 reports the distribution of bootstrap classification support, all cases in which the classification obtained from the averaged payoff matrix differs from the most frequent bootstrap classification, and, for each noise rate, the estimated share, bootstrap standard error, normal-approximation 95% confidence interval, and percentile bootstrap interval for the most prevalent game type in the unweighted and population-weighted analyses.

Robustness checks

Three robustness checks were conducted to assess whether the central qualitative patterns persist under alternative choices for the evolutionary horizon, strategy set, and payoff scale. First, the replicator dynamics were

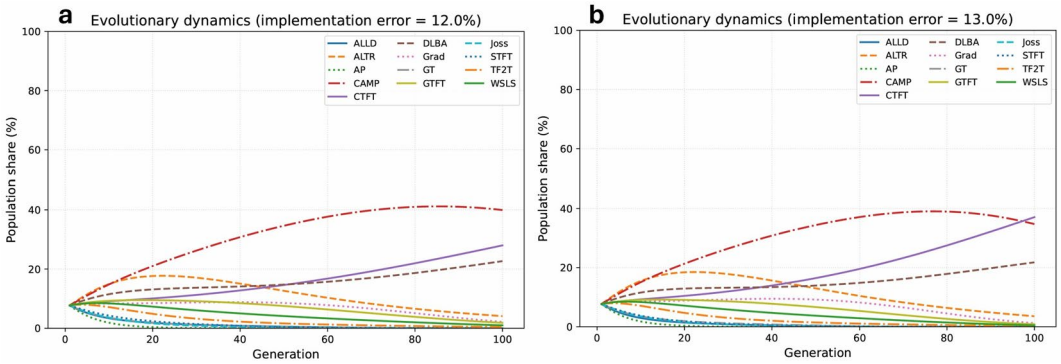


Fig. 6. Evolutionary dynamics at 12% and 13% implementation error. Population shares of the 13 strategies over 100 generations under discrete-time replicator dynamics for (**a**) 12% and (**b**) 13% negative implementation error (*CD*). Dynamics are based on payoff matrices estimated from 100 round-robin replicates per noise level, with expected match length 200 rounds (δ=0.995). Within each replicate, 100 self-play matches are simulated for each diagonal entry, with both players' payoffs assigned to the corresponding self-play payoff, and 200 matches are simulated for each cross-strategy pair. The x-axis shows the generation; the y-axis shows the population share in percent.

run for 500 rather than 100 generations. Second, the stochastic strategies GTFT and JOSS were excluded. Third, the tournament was repeated with an alternative payoff scale $(T, R, P, S) = (4, 3, 2, 1)$, instead of the baseline scale $(5, 3, 1, 0)$. The complete robustness-check results are provided in Supplementary Data 4.

Extending the evolutionary horizon to 500 generations increases population concentration, especially around CTFT at high noise levels. As a result, pairwise induced games involving CTFT receive greater weight in the evolved population. DLBA becomes the second-most prevalent strategy in most cases, replacing CAMP in the corresponding 100-generation results. Because CTFT becomes more prevalent under the longer horizon, Prisoner's Dilemma, or Deadlock at 11%, 13%, and 20% implementation error, becomes the most prevalent pairwise induced long-run game type at higher noise levels. The substitution pattern between Prisoner's Dilemma and Deadlock therefore remains visible. Because CAMP becomes rare under the longer horizon, Coordination no longer contributes substantially to the final-generation distribution. At lower noise levels, Prisoner's Dilemma and Chicken account for most of the pairwise induced game distribution, whereas Concord becomes less frequent, Stag Hunt largely disappears, and other game types become rare. Despite these quantitative changes, the population-weighted classifications remain concentrated on Concord and Harmony once implementation errors are introduced. However, the transition point shifts because CTFT becomes highly prevalent at lower noise levels under the longer evolutionary horizon.

Excluding the stochastic strategies GTFT and JOSS changes the quantitative distribution of induced game types and shifts some transition points. In this specification, Deadlock appears only at $\varepsilon = 19\%$, rather than at 11%, 13%, and 20% as in the full strategy set. Chicken reaches almost 40% at $\varepsilon = 1\%$, but then drops below 20% between $\varepsilon = 2\%$ and $\varepsilon = 5\%$, and declines thereafter. Conversely, the induced Prisoner's Dilemma becomes more frequent, reaching approximately 60–80% for $\varepsilon \geq 4\%$. Coordination remains present at some higher noise rates but is less prevalent and persistent than Prisoner's Dilemma, reaching approximately 20%. The population-weighted classifications again remain concentrated on Concord and Harmony, although the transition between them occurs at different noise rates because CTFT becomes the most prevalent strategy from $\varepsilon = 5\%$ onward.

Changing the payoff scale to $(T, R, P, S) = (4, 3, 2, 1)$ also shifts transition points and relative shares but does not change the main qualitative pattern. Deadlock appears at noise rates of 11%, 13%, and 17%, where Prisoner's Dilemma is largely absent. Chicken again reaches almost 40% but declines more steadily than under the baseline payoff scale. Outside the three Deadlock cases, the share of Prisoner's Dilemma generally lies between approximately 35% and 55% from $\varepsilon = 3\%$ onward, compared with approximately 20–40% in the baseline specification. Coordination appears from $\varepsilon = 10\%$ onward but remains below 20%. Under this specification, the population-weighted classifications also remain concentrated primarily on Concord and Harmony once negative implementation errors are introduced.

Overall, the robustness checks show that the exact shares of induced game types and the noise rates at which classifications switch depend on the strategy set, payoff scale, and evolutionary horizon. The qualitative conclusions nevertheless remain unchanged: negative implementation errors reshape the pairwise induced game distribution, the most prevalent strategies drive the evolved population structure, and the population-weighted classifications concentrate on a small set of game types.

Discussion

This section summarizes the main findings, discusses the scope and limitations of the analysis, and outlines directions for future research.

First, under noise-free play, the encounter-weighted share of induced Prisoner's Dilemma games is negligible in the evolved population because the strategies involved in these pairings become rare, given the strategy set and parameter values considered here. However, once implementation errors are introduced ($\varepsilon > 0$), Prisoner's

Dilemma emerges as an induced long-run game for a nontrivial share of strategy pairings, although the stage game remains unchanged. Second, after replicator updating, the induced games observed in the evolved population are shaped primarily by the most prevalent strategies, because these strategies determine which pairings receive substantial weight. Third, under population weighting, only a small subset of classifications is observed. Concord and Harmony prevail once implementation errors are introduced, whereas the noise-free case is classified as a "Tie" because many cooperative strategies earn nearly identical payoffs and small differences in population shares break the exact equality required for "Zero".

These findings illustrate the difference between pairwise induced game classifications and population-weighted payoff classifications. The former asks what type of 2×2 game is generated when two repeated game strategies are treated as the available choices. The latter asks which payoff ordering characterizes the evolved system after payoff entries are weighted by the strategies' population shares. The two classifications therefore need not coincide: a pairwise interaction may be classified as Prisoner's Dilemma, Deadlock, or Chicken, while the corresponding system-level structure may be Concord or Harmony.

These classifications depend on which strategies become prevalent under evolutionary selection. In the present simulations, a small core of high-frequency strategies generates most weighted matchups, whereas low-frequency strategies contribute little to the system-level classification. The transition between Concord and Harmony therefore reflects changes in the population composition, especially the relative prevalence of CTFT and CAMP under the baseline 100-generation specification. This also explains why changing the evolutionary horizon can alter the population-weighted classification: transient and later, more concentrated populations need not generate the same system-level structure.

Several limitations should be emphasized, and all results should be interpreted relative to the underlying parameter choices. First, the analysis considers only negative implementation errors, in which intended cooperation can be implemented as defection, while intended defection is always implemented correctly. In empirical settings, positive implementation errors, symmetric noise, or divergent perception[19,36] may also occur. The results should therefore be interpreted as a special case. Extending the analysis to positive and symmetric implementation errors, as well as to divergent perception, is an important direction for future work.

A second limitation concerns the evolutionary updating rule and the finite evolutionary horizon. The analysis uses deterministic discrete-time replicator dynamics, assuming an infinite population without mutation, but alternatives such as the Moran process[54] or shrink rate algorithms[36] could yield different population concentrations and thus different population-weighted classifications. The 500-generation robustness check further shows that longer evolutionary horizons can substantially affect population composition, changing both the encounter-weighted distribution of pairwise induced game types and the population-weighted classifications.

Another methodological limitation concerns the payoff classification itself. Classifying an induced game as Stag Hunt, Deadlock, or Concord identifies the ordinal structure of the long-run payoff matrix, but it does not imply that all games assigned to the same type are cardinally similar. Two induced games may both be classified as Stag Hunt while differing substantially in the size of their payoff gaps. Moreover, the main analysis classifies payoff matrices averaged across matches and tournament replicates before the induced game type is assigned. However, the bootstrap analysis indicates that this procedure is generally stable: the classification obtained from the averaged payoff matrix differed from the most frequent bootstrap classification in only 0.8% of cases across both analyses.

A further limitation concerns strategy selection. The 13 strategies considered here are canonical and cover several behavioral types. However, other strategy sets would also be possible, including TFT-ATFT[20,55], Omega-TFT, and Gradual-TFT[56], as well as zero-determinant (ZD) strategies[57,58]. The results should therefore be interpreted as applying to the canonical strategy set examined here.

Another limitation concerns the population-weighted classification. Because population-weighted payoff entries combine long-run payoffs with encounter frequencies, they form a system-level payoff representation rather than a game in the same sense as the pairwise induced 2×2 games. This distinction clarifies the sense in which the IPD can be said to "become" another game. At the stage game level, it does not: each round remains a Prisoner's Dilemma. At the induced game level, however, the long-run payoff ordering generated by pairs of repeated game strategies can correspond to a different 2×2 game type. Because its entries combine long-run payoffs with population shares, it is not a game in the same sense as the pairwise induced game, but a system-level payoff representation. Thus, whether the Prisoner's Dilemma "becomes" another game depends on the analytical perspective adopted.

Data availability

The game classification table is provided in Supplementary Data 1. The uncertainty and bootstrap results are provided in Supplementary Data 2. The baseline simulation results are provided in Supplementary Data 3, and the robustness-check results are provided in Supplementary Data 4. Additional information is available from the author upon reasonable request.

Code availability

The Python code has been deposited in Zenodo and is available at https://doi.org/10.5281/zenodo.21837336.

Received: 18 March 2026; Accepted: 15 September 2026
Published online: 21 September 2026

References

1. Axelrod, R. & Hamilton, W. D. The evolution of cooperation. *Science* **211**(4489), 1390–1396. https://doi.org/10.1126/science.7466396 (1981).
2. Boyd, R. & Lorberbaum, J. No pure strategy is evolutionarily stable in the repeated Prisoner's Dilemma game. *Nature* **327**, 58–59. https://doi.org/10.1038/327058a0 (1987).
3. Dal Bó, P. & Fréchette, G. R. The evolution of cooperation in infinitely repeated games: Experimental evidence. *Am. Econ. Rev.* **101**(1), 411–429. https://doi.org/10.1257/aer.101.1.411 (2011).
4. Glynatsi, N. E., Knight, V. & Harper, M. Properties of winning Iterated Prisoner's Dilemma strategies. *PLoS Comput. Biol.* **20**(12), e1012644. https://doi.org/10.1371/journal.pcbi.1012644 (2024).
5. Glynatsi, N. E., Akin, E., Nowak, M. A. & Hilbe, C. Conditional cooperation with longer memory. *PNAS* **121**(50), e2420125121. https://doi.org/10.1073/pnas.2420125121 (2024).
6. Hilbe, C., Nowak, M. A. & Sigmund, K. Evolution of extortion in iterated Prisoner's Dilemma games. *PNAS* **110**(17), 6913–6918. https://doi.org/10.1073/pnas.1214834110 (2013).
7. Le, S. & Boyd, R. Evolutionary dynamics of the continuous iterated Prisoner's dilemma. *J. Theor. Biol.* **245**(2), 258–267. https://doi.org/10.1016/j.jtbi.2006.09.016 (2007).
8. de Melo, C. M. & Terada, K. The interplay of emotion expressions and strategy in promoting cooperation in the iterated prisoner's dilemma. *Sci. Rep.* **10**, 14959. https://doi.org/10.1038/s41598-020-71919-6 (2020).
9. Stephens, D. W., McLinn, C. M. & Stevens, J. R. Discounting and reciprocity in an iterated Prisoner's Dilemma. *Science* **298**(5601), 2216–2218. https://doi.org/10.1126/science.1078498 (2002).
10. Abreu, D. & Rubinstein, A. The structure of Nash equilibrium in repeated games with finite automata. *Econometrica* **56**(6), 1259–1281. https://doi.org/10.2307/1913097 (1988).
11. Fudenberg, D. & Maskin, E. The folk theorem in repeated games with discounting or with incomplete information. *Econometrica* **53**(3), 533–554. https://doi.org/10.2307/1911307 (1986).
12. Axelrod, R. Effective choice in the Prisoner's Dilemma. *J. Conflict Resolut.* **24**(1), 3–25. https://doi.org/10.1177/002200278002400101 (1980).
13. Axelrod, R. More effective choice in the Prisoner's Dilemma. *J. Conflict Resolut.* **24**(3), 379–403. https://doi.org/10.1177/002200278002400301 (1980).
14. Axelrod, R. & Wu, J. How to cope with noise in the iterated Prisoner's Dilemma. *J. Conflict Resolut.* **39**(1), 183–189. https://doi.org/10.1177/002200279503900100 (1995).
15. Cheng, Z., Chen, G. & Hong, Y. Misperception influence on zero-determinant strategies in iterated Prisoner's Dilemma. *Sci. Rep.* **12**, 5174. (2022). https://doi.org/10.1038/s41598-022-08750-8
16. Kraines, D. & Kraines, V. Learning to cooperate with pavlov an adaptive strategy for the iterated prisoner's dilemma with noise. *Theor. Decis.* **35**(2), 107–150. https://doi.org/10.1007/BF01074955 (1993).
17. Rand, D. G., Ohtsuki, H. & Nowak, M. A. Direct reciprocity with costly punishment: Generous tit-for-tat prevails. *J. Theor. Biol.* **256**(1), 45–57. https://doi.org/10.1016/j.jtbi.2008.09.015 (2009).
18. Rapoport, A., Seale, D. A. & Coleman, A. Is tit-for-tat the answer? On the conclusions drawn from Axelrod's tournaments. *PLoS ONE* **10**(7), e0134128. https://doi.org/10.1371/journal.pone.0134128 (2015).
19. Signorino, C. S. Simulating international cooperation under uncertainty: The effects of symmetric and asymmetric noise. *J. Conflict Resolut.* **40**(1), 152–205. https://doi.org/10.1177/0022002796040001008 (1996).
20. Do Yi, S., Baek, S. K. & Choi, J. K. Combination with anti-tit-for-tat remedies problems of tit-for-tat. *J. Theor. Biol.* **412**, 1–7. https://doi.org/10.1016/j.jtbi.2016.09.017 (2017).
21. Nowak, M. A. & Sigmund, K. A strategy of win-stay, lose-shift that outperforms tit-for-tat in the Prisoner's Dilemma game. *Nature* **364**(6432), 56–58. https://doi.org/10.1038/364056a0 (1993).
22. Traulsen, A. & Glynatsi, N. E. The future of theoretical evolutionary game theory. *Philos. Trans. R. Soc. Lond. B Biol. Sci.* **378**, 20210508. https://doi.org/10.1098/rstb.2021.0508 (2023).
23. Nowak, M. A. Five rules for the evolution of cooperation. *Science* **314**, 1560–1563. https://doi.org/10.1126/science.1133755 (2006).
24. Taylor, C. & Nowak, M. A. Transforming the dilemma. *Evolution* **61**(10), 2281–2292. https://doi.org/10.1111/j.1558-5646.2007.00196.x (2007).
25. Su, Q., McAvoy, A., Wang, L. & Nowak, M. A. Evolutionary dynamics with game transitions. *PNAS* **116**(51), 25398–25404. https://doi.org/10.1073/pnas.1908936116 (2019).
26. Ito, H. & Yamamichi, M. A complete classification of evolutionary games with environmental feedback. *PNAS Nexus* https://doi.org/10.1093/pnasnexus/pgae455 (2024).
27. Shuvo, F. R. & Kabir, K. M. Investigating the impact of environmental feedback on the optional prisoner's dilemma for insights into cyclic dominance and evolution of cooperation. *R. Soc. Open Sci.* **11**(10), 240717. https://doi.org/10.1098/rsos.240717 (2024).
28. Feng, M., Pi, B., Deng, L.-J. & Kurths, J. An evolutionary game with the game transitions based on the Markov process. *IEEE Trans. Syst. Man Cybern. Syst.* **54**(1), 609–621. https://doi.org/10.1109/TSMC.2023.3315963 (2023).
29. Pi, B., Feng, M., Deng, L.-J., Chen, X. & Szolnoki, A. Evolutionary dynamics of variable games in structured populations. *IEEE Trans. Cybern. (Early Access)* https://doi.org/10.1109/TCYB.2026.3674776 (2026).
30. Blonski, M. & Spagnolo, G. Prisoners' other Dilemma. *Int. J. Game Theory* **44**(1), 61–81. https://doi.org/10.1007/s00182-014-0419-9 (2015).
31. Mengel, F., Orlandi, L. & Weidenholzer, S. Match length realization and cooperation in indefinitely repeated games. *J. Econ. Theory* **200**, 105416. https://doi.org/10.1016/j.jet.2022.105416 (2022).
32. Nowak, M. A. *Evolutionary Dynamics: Exploring the Equations of Life* (Harvard University Press, 2006). https://doi.org/10.2307/j.ctvjghw98.
33. Skyrms, B. *The Stag Hunt and the Evolution of Social Structure* (Cambridge University Press, 2004). https://doi.org/10.1017/CBO9781139165228.
34. Bruns, B. R. Names for games: Locating 2 × 2 games. *Games* **6**(4), 495–520. https://doi.org/10.3390/g6040495 (2015).
35. Murase, Y. & Baek, S. K. Five rules for friendly rivalry in direct reciprocity. *Sci. Rep.* **10**, 16904. https://doi.org/10.1038/s41598-020-73855-x (2020).
36. Herrera, Y. M. & Kydd, A. H. Don't look back in anger: Cooperation despite conflicting historical narratives. *Am. Political Sci. Rev.* **118**(3), 1–15. https://doi.org/10.1017/S0003055423001223 (2024).
37. Wedekind, C. & Milinski, M. Human cooperation in the simultaneous and the alternating Prisoner's Dilemma: Pavlov versus Generous Tit-for-Tat. *PNAS* **93**(7), 2686–2689. https://doi.org/10.1073/pnas.93.7.2686 (1996).
38. Axelrod, R. & Dion, D. The further evolution of cooperation. *Science* **242**(4884), 1385–1390. https://doi.org/10.1126/science.2424884.1385 (1988).
39. Harper, M. et al. Reinforcement learning produces dominant strategies for the Iterated Prisoner's Dilemma. *PLoS ONE* **12**(12), e0188046. https://doi.org/10.1371/journal.pone.0188046 (2017).
40. Li, J. How to design a strategy to win an IPD tournament. In *The Iterated Prisoner's Dilemma. 20 Years On* (eds Kendall, G. et al.) 89–104 (World Scientific, 2007). https://doi.org/10.1142/9789812770684_0004.
41. de Paulo, K. P., Estombelo-Montesco, C. A. & Tejada, J. New memory-one strategies of the iterated Prisoner's Dilemma: A new framework to programmed human–AI interaction. *Discov. Psychol.* **4**(1), 20. https://doi.org/10.1007/s44202-024-00133-6 (2024).

42. Press, W. H. & Dyson, F. J. Iterated Prisoner's Dilemma contains strategies that dominate any evolutionary opponent. *PNAS* **109**(26), 10409–10413. https://doi.org/10.1073/pnas.1206569109 (2012).
43. Burguillo, J. C. Probabilistic memory-one strategies to dominate the iterated Prisoner's Dilemma over networks. *Simul. Model. Pract. Theory* **119**, 102582. https://doi.org/10.1016/j.simpat.2022.102582 (2022).
44. Bruner, J. P. A note on the evolution of social learning. *Biol. Philos.* **40**(3), 11. https://doi.org/10.1007/s10539-025-09984-2 (2025).
45. Sugden, R. The economics of rights, co-operation and welfare. In *Softcover Reprint of the Hardcover*, 1st edn (Springer, 2005). https://doi.org/10.1057/9780230536791
46. Axelrod, R. *The Evolution of Cooperation. With a new Foreword by Richard Dawkins*. Rev. Ed. (Basic Books, 2006).
47. Mittal, S. & Deb, L. Optimal strategies of the iterated Prisoner's Dilemma problem for multiple conflicting objectives. *IEEE Trans. Evol. Comput.* **13**(3), 554–565. https://doi.org/10.1109/TEVC.2008.2009459 (2009).
48. Kim, M., Choi, J. K. & Baek, S. K. Win-Stay-Lose-Shift as a self-confirming equilibrium in the iterated Prisoner's Dilemma. *Proc. R. Soc. B Biol. Sci.* **288**, 1953. https://doi.org/10.1098/rspb.2021.1021 (2021).
49. Imhof, L. A., Fudenberg, D. & Nowak, M. A. Evolutionary cycles of cooperation and defection. *PNAS* **102**(31), 10797–10800. https://doi.org/10.1073/pnas.0502589102 (2005).
50. Hilbe, C., Nowak, M. A. & Traulsen, A. Adaptive dynamics of extortion and compliance. *PLoS ONE* **8**(11), e77886. https://doi.org/10.1371/journal.pone.0077886 (2013).
51. Beaufils, B., Delahaye, J. P. & Mathieu, P. Our meeting with gradual: A good strategy for the iterated prisoner's dilemma. In *Artificial Life V: Proceedings of the Fifth International Workshop on the Synthesis and Simulation of Living Systems* (eds Langton, C. G. & Shimohara, K.) 202–212 (Bradford Books, MIT Press, 1997).
52. Li, J., Hingston, P. & Kendall, G. Engineering design of strategies for winning iterated Prisoner's Dilemma competitions. *IEEE Trans. Comput. Intell. AI Games* **3**(4), 348–360. https://doi.org/10.1109/TCIAIG.2011.2166268 (2011).
53. Snyder, G. H. & Diesing, P. *Conflict Among Nations* (Princeton University Press, 2015). https://doi.org/10.1515/9781400871186.
54. Moran, P. A. P. Random processes in genetics. *Math. Proc. Camb. Philos. Soc.* **54**(1), 60–71. https://doi.org/10.1017/S0305004100033193 (1958).
55. Murase, Y. & Baek, S. K. Automata representation of successful strategies for social dilemmas. *Sci. Rep.* **10**, 13370. https://doi.org/10.1038/s41598-020-70281-x (2020).
56. Su, Q. et al. A multi-agent reinforcement learning framework for exploring dominant strategies in iterated and evolutionary games. *Nat. Commun.* **17**, 490. https://doi.org/10.1038/s41467-025-67178-6 (2026).
57. Hilbe, C., Wu, B., Traulsen, A. & Nowak, M. A. Evolutionary performance of zero-determinant strategies in multiplayer games. *J. Theor. Biol.* **374**, 115–124. https://doi.org/10.1016/j.jtbi.2015.03.032 (2015).
58. Ueda, M. Memory-two zero-determinant strategies in repeated games. *R. Soc. Open Sci.* **8**, 202186. https://doi.org/10.1098/rsos.202186 (2021).

Acknowledgements

I am grateful to the reviewers for their constructive comments and suggestions. I also thank friends and colleagues for helpful feedback, encouragement, and support, especially Ralph Rotte, Jorge Peña, Kevin Lu, Finn Gebbensleben, Athena Riazsadri, and Corinna Lenz.

Author contributions

A.D. designed the research and wrote the paper.

Funding

Open Access funding enabled and organized by Projekt DEAL. The author declares that no funding was received for this work.

Declarations

Competing interests

The author declares no competing financial interests.

Additional information

Supplementary Information The online version contains supplementary material available at https://doi.org/10.1038/s41598-026-72270-y.

Correspondence and requests for materials should be addressed to A.D.

Reprints and permissions information is available at www.nature.com/reprints.

Publisher's note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.

Open Access This article is licensed under a Creative Commons Attribution 4.0 International License, which permits use, sharing, adaptation, distribution and reproduction in any medium or format, as long as you give appropriate credit to the original author(s) and the source, provide a link to the Creative Commons licence, and indicate if changes were made. The images or other third party material in this article are included in the article's Creative Commons licence, unless indicated otherwise in a credit line to the material. If material is not included in the article's Creative Commons licence and your intended use is not permitted by statutory regulation or exceeds the permitted use, you will need to obtain permission directly from the copyright holder. To view a copy of this licence, visit http://creativecommons.org/licenses/by/4.0/.

© The Author(s) 2026