GRAPHZD: GRAPH-AWARE CONTINUOUS ZERO-DETERMINANT CONTROL IN EVOLUTIONARY NETWORKS

Anonymous authors
Paper under double-blind review

ABSTRACT

The emergence and maintenance of cooperation in finite, structured populations remain a fundamental challenge in evolutionary game theory. Zero-determinant (ZD) strategies can enforce linear relations between long-term payoffs, yet fixed extortionate or generous control cannot adapt to heterogeneous neighborhood conditions. This limitation is particularly important in evolutionary networks, where local structure shapes both interaction payoffs and strategy replacement. In this paper, we propose GraphZD, a graph-aware continuous ZD controller for repeated Prisoner's Dilemma games on evolutionary networks. The method instantiates a graph-aware continuous relational-control paradigm that treats the enforceable payoff relation, rather than a predefined strategy label, as the object of adaptation. A graph attention encoder summarizes local behavior, payoff, and structure. An online contextual linear controller evaluates nine control anchors and forms a continuous proposal, which is regulated by an identification phase, exponential moving-average smoothing, and a coexistence gate. An analytically computed common scale then maps the control to a feasible memory-one multiplicative ZD policy without coordinate-wise clipping, preserving the intended Press–Dyson vector direction. We evaluate GraphZD over 5,000 death–birth generations on grid, scale-free, SBM, and small-world networks initialized with 20% adaptive carriers and 80% AllD. In paired experiments, GraphZD improves population share, cooperation, and collective payoff over an otherwise identical controller without graph information. It also outperforms fixed extortionate and generous ZD endpoints: relative to the stronger generous endpoint, GraphZD raises mean population share by 17.8 percentage points and mid-run social welfare by 0.84. The trajectories support a two-stage interpretation: graph-aware control improves survival under early defector pressure, allowing cooperative groups to persist and generate higher social welfare later. These results connect local structural information with adaptive payoff control in finite evolutionary populations.

1 INTRODUCTION

The emergence and maintenance of cooperation in finite populations are central problems in repeated and evolutionary games Axelrod & Hamilton (1981); Nowak (2006). Free-riders obtain benefits from cooperative behavior without bearing the corresponding cost, so cooperation is difficult to sustain under well-mixed interaction. Network structure changes this pressure: spatial clustering can protect cooperative groups on regular graphs, while degree heterogeneity creates influential hubs in scale-free networks Nowak & May (1992); Ohtsuki et al. (2006); Santos & Pacheco (2005). Zero-determinant (ZD) strategies offer a complementary mechanism. In an iterated game, a memory-one player[1] can enforce a linear relation between its long-run payoff and that of its opponent Press & Dyson (2012).

Fixed ZD strategies nevertheless face an evolutionary tradeoff. Extortionate policies can resist exploitation and gain a local fitness advantage, but sustained extortion suppresses cooperation and collective welfare. Generous policies support reciprocal cooperation, yet they may be weak when rare

[1]We use "player" and "agent" interchangeably for an IPD participant; "node" refers to its position in the interaction graph.

Reviewers: please read the Reviewer Guidelines (iclr.cc/Conferences/2027/ReviewerGuidelines) and the AI Policy for Reviewers (iclr.cc/Conferences/2027/AIPolicyForReviewers). If you used AI to expand, edit, or polish your review, please provide the input text to the LLM. Better still, consider skipping the LLM and submitting your original text: we, and the authors, are much more interested in your unedited thoughts than in what an LLM has to say. AI-assisted or not, you are putting your name and reputation behind your review: LLM-generated falsehoods, hallucinations or misrepresentations are subject to disciplinary action, which may include desk-rejecting all papers you have authored.

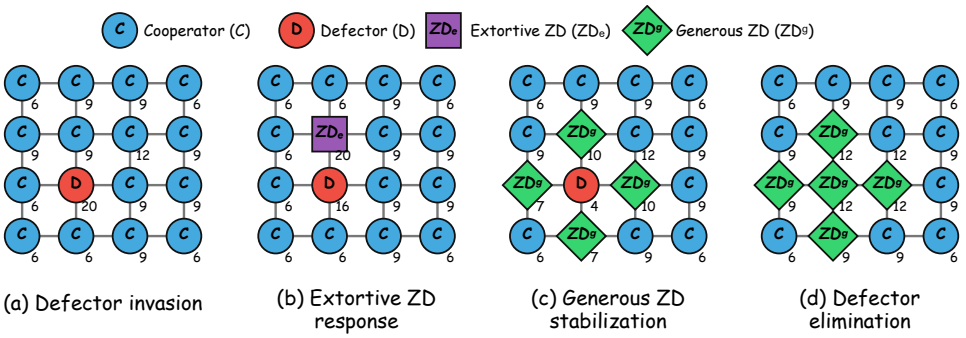


Figure 1: Conceptual motivation for structure-aware ZD control. A defector creates an exposed low-welfare boundary; an extortion-oriented response can resist local exploitation, whereas a generosity-oriented response can stabilize cooperation after cooperative support expands. The figures and payoff annotations are illustrative rather than simulation snapshots, and ZD_e and ZD_g denote the two ends of the continuous control spectrum.

or exposed to defectors Hilbe et al. (2013); Adami & Hintze (2013). Neither endpoint is therefore appropriate throughout an invasion process: early survival and later cooperative consolidation place different demands on the policy.

This tradeoff becomes harder in structured populations because evolutionary outcomes depend on network position and neighborhood composition. Players with similar recent payoffs may occupy very different network positions: one may lie inside a cooperative cluster, while another is exposed at a boundary dominated by defectors. A controller based only on individual payoff history cannot reliably distinguish them. At the same time, adapting a ZD policy requires more than interpolating or independently clipping four cooperation probabilities, because such operations may destroy the intended payoff relation. Figure 1 summarizes this motivation: extortion- and generosity-oriented behavior form the two sides of a continuous control, while local graph context informs when the response should change.

In this paper, we propose **GraphZD** (**Graph**-aware continuous **Z**ero-**D**eterminant control), which maps local observations to continuous, feasible payoff control. The method instantiates a broader graph-aware continuous relational-control paradigm. Under this paradigm, local graph context is used to continuously regulate an enforceable payoff relation, rather than to select a predefined strategy or directly optimize unconstrained cooperation probabilities. A one-hop graph neural network (GNN), implemented as a graph attention network (GAT), encodes local behavioral, payoff, and structural observations. A shared online contextual linear controller scores nine anchors using welfare, advantage over the always-defect strategy (AllD), and post-replacement neighborhood feedback, then forms a continuous proposal from their softmax-weighted expectation. A bounded identification phase, exponential moving-average (EMA) smoothing, and a coexistence gate regulate the proposal over time. Finally, uniform feasible scaling maps the resulting control to a memory-one multiplicative ZD policy, preserving the intended Press–Dyson direction for every instantiated policy. Our experiments isolate the value of graph context, trace invasion and welfare, and compare continuous adaptation with fixed ZD endpoints under adverse initial conditions.

Our contributions can be summarized as follows:

- We introduce a graph-aware continuous relational-control paradigm that uses local graph context to adapt an enforceable payoff relation rather than selecting among fixed ZD strategies.
- We derive a feasible multiplicative ZD map that uses one common scale, avoids coordinate-wise clipping, and preserves the Press–Dyson direction for every control in $[-1, 1]$.
- GraphZD reaches fixation in every run. Against *w/o GNN*, mean share, mid-run cooperation, and mid-run social welfare improve by 2.9 and 11.8 percentage points and 0.451, respectively; majority is reached 66 generations sooner. Against ZD-GEN, the gains are 17.8 and 21.6 points and 0.842. This links early survival to later cooperative consolidation and welfare.

2 Related Work

Zero-determinant payoff control. Press & Dyson (2012) established that a memory-one strategy can enforce a linear relation between long-run payoffs. Subsequent work showed that extortion, generosity, and evolutionary success need not align: a policy that performs well in a pairwise encounter may be disfavored under population replacement, while generous variants can support cooperation without being uniformly robust when rare Hilbe et al. (2013); Adami & Hintze (2013). More recent constructions extend ZD control to networked, asymmetric, noisy, and finite games Cheng & Li (2024); Kang et al. (2024); Zhang et al. (2025). The dynamically switching PERG strategy also responds to population conditions (Huang et al., 2026), but switches among prescribed modes. GraphZD instead treats extortion and generosity as endpoints of a continuous feasible family and adapts the enforced payoff direction for the player at each node.

Evolutionary games on networks. Population structure determines which players interact and which neighboring nodes compete for replacement. Spatial clustering can protect cooperation on regular graphs, whereas heterogeneous degree and small-world connectivity produce different assortment and propagation patterns Nowak & May (1992); Abramson & Kuperman (2001); Ohtsuki et al. (2006); Santos & Pacheco (2005); Watts & Strogatz (1998). Evolutionary outcomes may also depend on a mutant's position and on symmetries of the update process McAvoy & Hauert (2015). Coevolutionary models alter links or interaction timing together with behavior Fu et al. (2009); Pacheco et al. (2006); Meng et al. (2025). Our setting keeps the graph fixed and adapts the policy instead, so local topology influences decisions without becoming another evolving state variable.

Graph representations and receptive field. Message-passing GNNs encode a node through repeated aggregation over its neighborhood Veličković et al. (2018); Wu et al. (2020). Increasing depth expands the nominal receptive field, but it can also blur local contrasts or compress long-range information. Recent work addresses these trade-offs through locality-aware and probabilistic graph rewiring Barbero et al. (2024); Qian et al. (2024), while theoretical analyses relate generalization to the structural variation captured by graph embeddings Li et al. (2025). Complementary work develops local structural positional encodings and synthetic-prior pretraining for cross-graph generalization Amboage et al. (2026); Choi et al. (2026). GraphZD uses a frozen one-hop encoder because the immediate neighborhood defines death–birth competition; the two-hop and online-training variants are evaluated separately rather than assumed to be superior.

Contextual and graph-aware bandits. Contextual bandits support sequential decisions from partial feedback Li et al. (2010); Lattimore & Szepesvári (2020), and graph neural bandits use relational representations to share information across structured users or actions Kassraie et al. (2022); Qi et al. (2023). Neural online regression further broadens the class of reward models available to contextual bandits Deb et al. (2024). These analyses typically assume an exogenous environment or a prescribed reward process. In GraphZD, actions affect payoffs, fitness, replacement, and hence future contexts. We therefore use a contextual linear controller as an online estimation mechanism and restrict the formal guarantee to its conditional surrogate, rather than importing a standard regret claim into the evolutionary process.

Research gap: Despite significant advances in ZD strategies, structured population dynamics, and GNN-Bandit methods, these three lines of research have evolved largely in isolation. ZD strategies possess strong theoretical guarantees but lack spatial awareness and adaptability; structured-population studies reveal the importance of network topology but rely on static rules or simple heuristics; GNN-Bandit frameworks offer powerful learning tools but have not addressed the specific challenges of evolutionary strategy adaptation. **GraphZD bridges this gap** by being the first framework to simultaneously (i) perceive network context through GNN, (ii) adapt ZD parameters via theoretically grounded bandit selection, and (iii) execute actions through the tractable ZD family.

3 Problem Setting

In an Iterated Prisoner's Dilemma (IPD), two players interact repeatedly by choosing between cooperation (C) and defection (D) in each round. From the focal player's perspective, the stage-game

payoffs corresponding to the four possible joint outcomes—mutual cooperation (CC), unilateral cooperation (CD), unilateral defection (DC), and mutual defection (DD)—are denoted by R, S, T, and P, respectively. Here, R represents the reward for mutual cooperation, S the sucker's payoff, T the temptation payoff for unilateral defection, and P the punishment for mutual defection. Standard IPD formulations require the condition $T > R > P > S$, along with $2R > T + S$ to establish mutual cooperation as the Pareto optimal outcome in repeated settings. In our work, we adopt the standard canonical payoff parametrization $(R, S, T, P) = (3, 0, 5, 1)$, which naturally satisfies $5 > 3 > 1 > 0$ and $2(3) > 5 + 0$.

We study repeated games and strategy replacement on a fixed interaction network represented by a connected, undirected graph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$ with $N = |\mathcal{V}|$. In our experiments, each node is populated by a player executing either the GraphZD or AllD strategy. Players occupying adjacent nodes play an IPD with the stage-game payoffs defined above. Within this framework, a memory-one policy is specified by the vector $\mathbf{p} = (p_{CC}, p_{CD}, p_{DC}, p_{DD})$, where each component $p_k \in [0, 1]$ denotes the probability of cooperating conditioned on the outcome of the previous pairwise interaction.

The evolutionary process has two time scales. Within each generation, the players connected by every edge play a $T_{\mathrm{IPD}} = 20$-round IPD. The pairwise $CC/CD/DC/DD$ state is maintained across these rounds but reset at the next generation. In contrast, behavioral summaries associated with each node—including the resident player's cooperation history, recent payoff history, and previous action—are retained across generations and used to construct the GNN and controller context. After the interaction stage, payoffs are aggregated, one death–birth replacement is performed, and the online controller statistics are updated. This distinction between behavioral and evolutionary time also separates actions, memory-one policies, and inherited strategy types: a player using GraphZD may defect in a round without becoming an AllD player, while replacement changes the strategy type at one node. The static network remains fixed throughout.

Fitness measures mean interaction performance so that high degree alone does not increase reproductive weight through accumulated payoff. Let $\pi_i^{(g,t)}(j)$ be the stage payoff received by the player at node i from the player at neighboring node j in round t of generation g. The mean interaction payoff and fitness of the player at node i are

$$\bar{\pi}_i(g) = \frac{1}{T_{\mathrm{IPD}} d_i} \sum_{j \in \mathcal{N}(i)} \sum_{t=1}^{T_{\mathrm{IPD}}} \pi_i^{(g,t)}(j), \qquad F_i(g) = \bar{\pi}_i(g) + 0.1,$$

where $d_i = |\mathcal{N}(i)|$. Thus, fitness is based on the mean payoff per interaction rather than the accumulated payoff of players at high-degree nodes. The additive constant ensures positive fitness.

Evolution follows a finite-population death–birth process related to the Moran process Moran (1958); Nowak (2006). A node v is selected for death, and a neighbor u fills the vacancy with probability

$$\Pr(u \to v \mid v) = \frac{F_u(g)}{\sum_{j \in \mathcal{N}(v)} F_j(g)}.$$

This defines a strong payoff-proportional selection rule. The offspring inherits the parent's strategy type and designated behavioral summaries. Pairwise outcome memory is reset, whereas the shared controller posterior is retained and the offspring's policy is subsequently recomputed from the context of its new node.

Initially, players at 20 randomly selected nodes use GraphZD, while those at the remaining 80 nodes use AllD. This defector-dominated initialization tests whether GraphZD can increase in prevalence while sustaining cooperation and population payoff over a finite horizon. We study finite-time invasion, cooperation, and payoff trajectories rather than evolutionary stability, infinite-population equilibria, or guaranteed fixation. When used, "emergence" denotes an empirical coordinated increase in GraphZD prevalence, cooperation, and population payoff in this finite system; it does not imply a thermodynamic phase transition or a topology-independent critical threshold.

4 GRAPHZD

GraphZD converts neighborhood observations into feasible pairwise policies and evaluates their consequences through local evolutionary competition (Figure 2). For each GraphZD player at

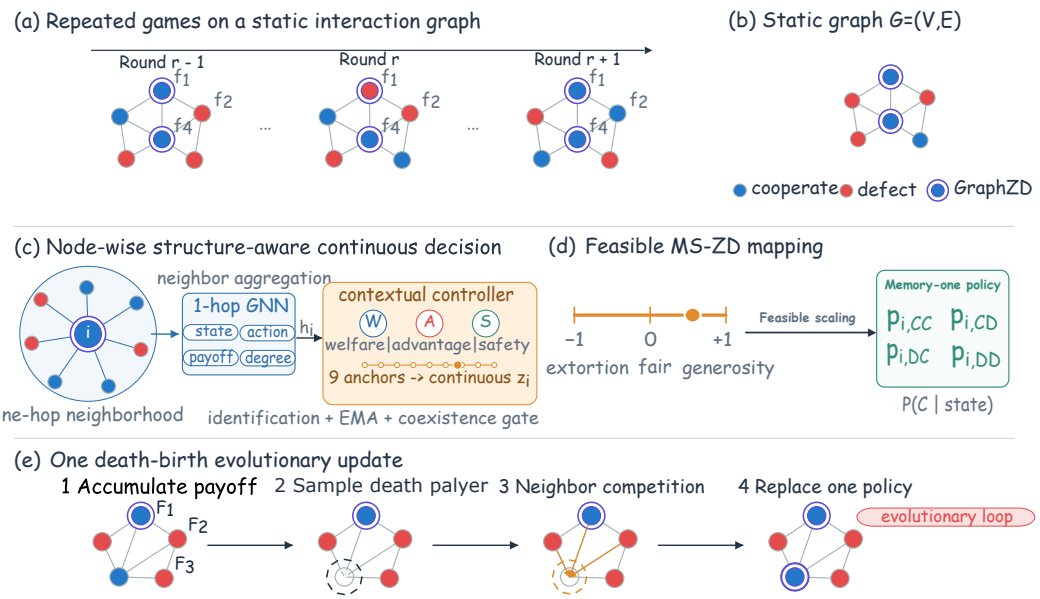


Figure 2: **Overview of GraphZD on a static evolutionary graph.** **(a)** Repeated pairwise games unfold over successive rounds on a fixed topology; actions, interaction histories, and payoffs can change while edges remain unchanged. **(b)** The graph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$ defines both interaction and replacement neighborhoods. Blue and red fills illustrate cooperation and defection, respectively, whereas purple outlines identify GraphZD carriers; actions and strategy types are distinct. **(c)** A shared one-hop GNN supplies node-specific structural context to an online contextual controller. Welfare, advantage relative to AllD, and post-replacement neighborhood feedback inform value estimates at nine anchors. Their softmax-weighted expectation produces a continuous proposal, supplemented by an identification phase, EMA smoothing, and a coexistence gate. **(d)** Uniform feasible scaling maps the resulting control to a valid memory-one multiplicative ZD (MS-ZD) policy. CC, CD, DC, and DD denote conditioning outcomes from the focal player's perspective, not a prescribed transition cycle; the triangular feasible region is schematic rather than computed. **(e)** After payoff accumulation and normalization, a death–birth event selects a replacement target uniformly and a reproducing neighbor in proportion to fitness. One strategy is replaced while the topology is preserved. Network states and replacement events are illustrative.

node i, a shared one-hop GNN produces a node-specific representation. The online controller uses it to estimate welfare, advantage relative to AllD, and post-replacement neighborhood retention, sharing posterior statistics while conditioning the player's decision on the context of node i. A softmax-weighted average of nine anchors produces the continuous control $z_i \in [-1, 1]$; bounded identification, coexistence gating, and EMA regulate its evolution, with a fair fallback during initialization or at low prevalence. Uniform feasible scaling maps it to the memory-one probabilities $\mathbf{p}_i = (p_{i,CC}, p_{i,CD}, p_{i,DC}, p_{i,DD}) \in [0, 1]^4$. Each component conditions on the previous outcome of the corresponding pairwise game. The mapping preserves the intended multiplicative ZD relation without independently clipping the four probabilities. Payoffs generated by these policies subsequently inform fitness-proportional neighbor competition in the death–birth process. This mechanism specifies how adaptive policies enter evolutionary selection; it does not itself guarantee cooperation, fixation, or evolutionary stability.

4.1 Graph Context Encoder

The encoder summarizes local behavior, payoff, and topology using historical cooperation, normalized payoff, normalized degree, and the most recent action. Appendix A.3 gives the exact input and normalization definitions.

A two-layer graph-attention checkpoint Veličković et al. (2018), following the broader message-passing GNN paradigm Wu et al. (2020), maps these features to a 16-dimensional embedding. To enforce a one-hop receptive field, the first GAT layer aggregates neighbors and the second layer uses only self-loops:

$$\mathbf{h}_i(g) = \mathrm{GAT}^{\mathrm{self}}_{\omega,2}\left(\mathrm{ReLU}\left(\mathrm{GAT}_{\omega,1}(\mathbf{f}_i, \{\mathbf{f}_j : j \in \mathcal{N}(i)\})\right)\right).$$

For the confirmatory experiments, ω is pretrained on synthetic graphs and held fixed; Appendix A gives the training protocol. The normalized embedding is concatenated with an intercept, the current GraphZD population share, and normalized generation to form the 19-dimensional controller context. Freezing keeps these coordinates stationary when carriers are rare and feedback is sparse, while the one-hop restriction matches death–birth parent selection. Appendices A.3, C.1, and C.4 give the exact context, two-hop comparison, and online-training variant.

4.2 Continuous contextual controller

The controller learns how continuous payoff control affects local outcomes in each observed context. Let $z \in [-1, 1]$ denote the control, with negative values extortion-oriented, zero fair, and positive values generosity-oriented. We use the action-dependent feature

$$\boldsymbol{\psi}(\mathbf{x}, z) = [\mathbf{x},\ z\mathbf{x},\ z^2\mathbf{x}] \in \mathbb{R}^{57}. \tag{1}$$

The shared controller maintains one regularized design matrix $\mathbf{A}$ and three reward vectors $\mathbf{b}_w, \mathbf{b}_A, \mathbf{b}_\ell$. Their ridge estimates are $\widehat{\boldsymbol{\theta}}_q = \mathbf{A}^{-1}\mathbf{b}_q$ for welfare $q = w$, fitness advantage $q = A$, and post-replacement neighborhood retention $q = \ell$.

Three normalized feedback signals balance productive interaction against evolutionary persistence: local welfare, fitness advantage over defectors, and post-replacement neighborhood retention. Retention is observed only after replacement and is not an action-time GNN input. Their exact definitions are given in Appendix A.3; their weighted combination is examined through ablation and diagnostics.

Anchor evaluation converts these value estimates into a continuous proposal using an upper confidence bound (UCB). At decision time, nine candidates $\mathcal{Z} = \{-1, -0.75, \ldots, 1\}$ are scored by

$$U_i(z) = \boldsymbol{\psi}(\mathbf{x}_i, z)^\top(\widehat{\boldsymbol{\theta}}_w + \widehat{\boldsymbol{\theta}}_A + 0.5\widehat{\boldsymbol{\theta}}_\ell) + \alpha\sqrt{\boldsymbol{\psi}(\mathbf{x}_i, z)^\top\mathbf{A}^{-1}\boldsymbol{\psi}(\mathbf{x}_i, z)}. \tag{2}$$

Instead of a hard arm choice, GraphZD forms a soft proposal

$$\widetilde{z}_i = \sum_{z\in\mathcal{Z}} z\,\frac{\exp(U_i(z)/\tau)}{\sum_{z'\in\mathcal{Z}}\exp(U_i(z')/\tau)}.$$

The anchors support value estimation, while their softmax-weighted average produces a continuous proposal within their convex hull. The quadratic feature in Equation (1) allows the preferred control to vary with context while keeping prediction linear in the learned parameters. The UCB term favors uncertain anchors, and temperature τ determines how strongly anchors influence the proposal.

Initialization and temporal smoothing regulate how the proposal becomes an executed control. The first two controller horizons use $z = 0$ to collect feedback before departing from fairness. During generations $2H$ through $6H$, carrier-stratified probes in $\{-0.5, 0, 0.5\}$ sample negative, neutral, and positive controls. Thereafter, the proposal is multiplied by the coexistence gate $1 - s_g^8$ and smoothed:

$$z_i(g) = (1-\eta)z_i(g-1) + \eta(1 - s_g^8)\widetilde{z}_i(g).$$

EMA limits rapid reversals after noisy observations, while the coexistence gate attenuates the proposal near fixation. If population share falls below the emergency threshold, the controller returns to $z = 0$. The frozen configuration uses $H = 25$, $\alpha = 0.25$, $\tau = 0.1$, and $\eta = 0.2$; these components are held fixed across structural variants so that their comparison tests the representation.

4.3 Feasible multiplicative ZD mapping

A common feasible scale converts the scalar control into valid cooperation probabilities while preserving its Press–Dyson direction. For focal and opponent payoff vectors $\mathbf{s}_X = (R, S, T, P)$ and

Algorithm 1: GraphZD Evaluation Loop

Input : $\mathcal{G}$; frozen encoder $\omega^\star$; initial types; generations G; IPD rounds T; anchors $\mathcal{Z}$
Output : Trajectories $\mathcal{T}$

Initialize controller $(\mathbf{A}, \mathbf{b}_w, \mathbf{b}_A, \mathbf{b}_\ell)$, controls $\{z_i\}$, histories $\mathcal{H}$, and $\mathcal{T} \leftarrow \emptyset$;
for $g \leftarrow 0$ **to** $G-1$ **do**
  $\mathbf{X}_g \leftarrow \textsc{BuildFeatures}(\mathcal{G}, \mathcal{H})$;
  $\mathbf{H}_g \leftarrow \mathrm{GNN}_{\omega^\star}(\mathcal{G}, \mathbf{X}_g)$;
  $\mathcal{C}_g \leftarrow \{i : \mathrm{type}(i) = \mathrm{GraphZD}\}$;
  foreach $i \in \mathcal{C}_g$ **do**
    $\mathbf{x}_i \leftarrow \textsc{Context}(\mathbf{h}_i, g, |\mathcal{C}_g|/N)$;
    $q_{ik} \leftarrow \textsc{Score}(\mathbf{x}_i, z_k)$ for $z_k \in \mathcal{Z}$;
    $\widetilde{z}_i \leftarrow \sum_k \mathrm{softmax}(q_{ik}) z_k$;
    $z_i \leftarrow \mathrm{EMA}(\textsc{Gate}(\textsc{Identify}(\widetilde{z}_i)))$;
    $\mathbf{p}_i \leftarrow \mathbf{o} + 0.95\phi_{\max}(z_i)\mathbf{d}(z_i)$;
  $(\boldsymbol{\pi}, \mathcal{H}) \leftarrow \textsc{PlayIPD}(\mathcal{G}, \{\mathbf{p}_i\}, T, \mathcal{H})$;
  $\mathbf{F} \leftarrow \textsc{Fitness}(\boldsymbol{\pi})$;
  $v \sim \mathrm{Unif}(\mathcal{V}), u \sim \mathrm{Cat}\Big(\{F_j / \sum_{k \in \mathcal{N}(v)} F_k\}_{j \in \mathcal{N}(v)}\Big)$;
  $\mathrm{type}(v) \leftarrow \mathrm{type}(u)$;
  $\mathbf{y}_i \leftarrow (w_i, A_i, \ell_i)$ for $i \in \mathcal{C}_g$;
  $(\mathbf{A}, \mathbf{b}_w, \mathbf{b}_A, \mathbf{b}_\ell) \leftarrow \textsc{Update}(\{\mathbf{x}_i, z_i, \mathbf{y}_i\}_{i \in \mathcal{C}_g})$;
  $\mathcal{T} \leftarrow \mathcal{T} \cup \textsc{Measure}(g, \boldsymbol{\pi}, \mathcal{H})$;
return $\mathcal{T}$;

$\mathbf{s}_Y = (R, T, S, P)$, let

$$\chi(z) = 1 + |z|(\chi_{\max} - 1), \qquad \ell(z) = \begin{cases} P, & z \le 0, \\ R, & z > 0. \end{cases}$$

The Press–Dyson direction is

$$\mathbf{d}(z) = \mathbf{s}_X - \chi(z)\mathbf{s}_Y + (\chi(z) - 1)\ell(z)\mathbf{1}. \tag{3}$$

With offset $\mathbf{o} = (1, 1, 0, 0)$, define

$$\phi_{\max}(z) = \min\left(\left\{ \frac{1 - o_j}{d_j(z)} : d_j(z) > 0 \right\} \cup \left\{ \frac{-o_j}{d_j(z)} : d_j(z) < 0 \right\} \right). \tag{4}$$

Coordinates with $d_j(z) = 0$ impose no constraint on the common scale and are therefore excluded from the two sets. GraphZD uses $\phi(z) = 0.95\phi_{\max}(z)$ and

$$\mathbf{p}(z) = \mathbf{o} + \phi(z)\mathbf{d}(z). \tag{5}$$

For the stationary distribution induced by a memory-one interaction, the Press–Dyson direction corresponds to the linear relation

$$\pi_X - \ell(z) = \chi(z)\big(\pi_Y - \ell(z)\big),$$

whenever the usual determinant construction is well-defined. Negative and positive values of z select different baselines and thus the extortion- and generosity-oriented sides of the family; $z = 0$ gives the fair point. The 0.95 factor stays inside the boundary of the feasible region for numerical margin. It scales every coordinate by the same $\phi(z)$. Independent clipping would generally change coordinate ratios in $\mathbf{d}(z)$ and no longer preserve the stated linear relation.

Proposition 1 (Feasibility and direction preservation)**.** *For every $z \in [-1, 1]$, Equation* (5) *lies in $[0, 1]^4$. Moreover, $\mathbf{p}(z) - \mathbf{o}$ remains a scalar multiple of the Press–Dyson direction in Equation* (3)*.*

The proof is given in Appendix D.1.

GraphZD has two separate stages. Algorithm 2 learns the graph representation without using confirmatory trajectories; its selected encoder is then frozen in the evolutionary evaluation of Algorithm 1.

Proposition 1 directly guarantees probability validity and direction preservation for the executed policy construction. The conditional linear-UCB analysis and excitation diagnostics are deferred to Appendices D.3 and B.3; they do not imply convergence of the full controller or evolutionary success.

Table 1: Overall differences across 90 runs. Brackets give paired-bootstrap 95% confidence intervals; all Holm-adjusted paired-t tests have $p_H < 0.001$.

Comparison	Mean population share	Mid cooperation	Mid social welfare
1-hop − w/o GNN	0.029 [0.015, 0.045]	0.118 [0.081, 0.159]	0.451 [0.304, 0.615]
1-hop − Fixed ZD-EX	0.583 [0.534, 0.630]	0.916 [0.903, 0.929]	3.652 [3.598, 3.701]
1-hop − Fixed ZD-GEN	0.178 [0.132, 0.227]	0.216 [0.161, 0.275]	0.842 [0.623, 1.081]

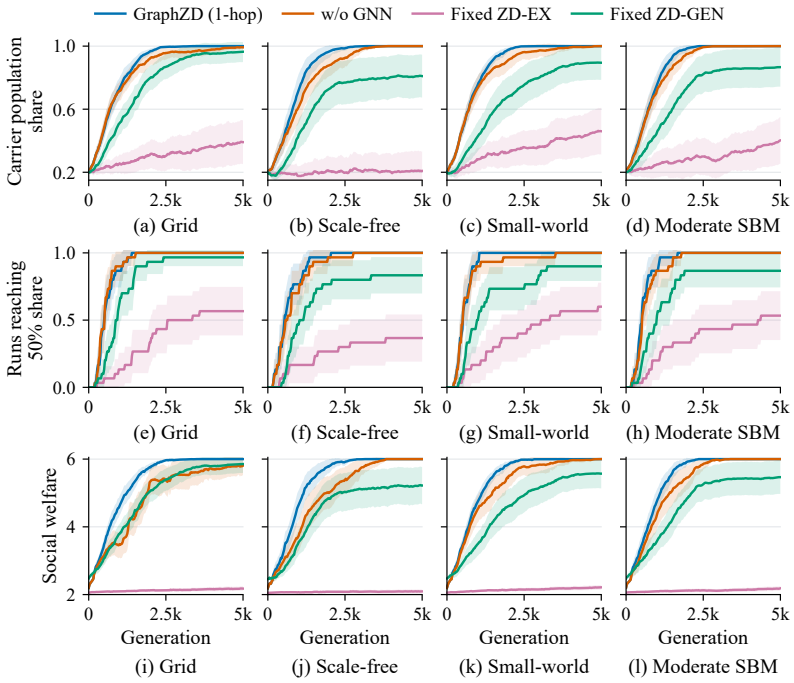


Figure 3: Trajectories over 30 seeds per topology. Rows show mean carrier population share, cumulative majority attainment, and social welfare; shading gives 95% normal confidence intervals. Appendix C.3 reports all three SBM networks.

5 Experiments

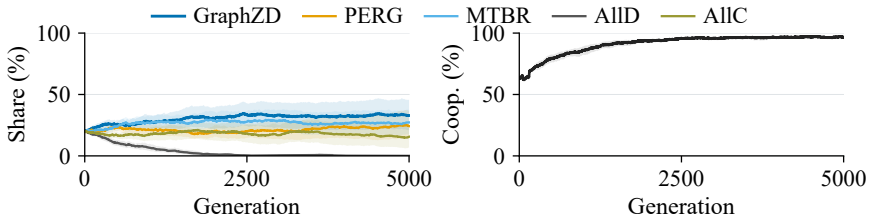
We first specify the experimental setting and then report each main result with its corresponding figure or table. The evaluation moves from confirmatory invasion dynamics to paired method differences and finally to an exploratory mixed-population competition. Seed-level visualizations and additional analyses are reported in the appendices.

5.1 Experimental Setup

Game parameters. We compare GraphZD[2], **w/o GNN**, **Fixed ZD-EX**, and **Fixed ZD-GEN**. Each run assigns adaptive or fixed-ZD players to 20 nodes and AllD players to the remaining 80 nodes of one of three $N = 100$ networks (Appendix C.2 extends the analysis to $N = 400$ and $N = 900$ and compares one- and two-hop encoders across population sizes): a periodic 10×10 Grid, a Barabási–Albert scale-free network ($m = 2$), or a Watts–Strogatz small-world network ($k = 4$, $p = 0.3$). Runs last 5,000 generations with 20 IPD rounds per edge. Each method is evaluated in 90 runs (30 seeds on each topology), giving 360 method runs in total.

Baseline. *Fixed ZD-EX* and *Fixed ZD-GEN* are fixed controls that set the continuous parameter to the extortionate and generous endpoints, respectively, without online estimation, identification, EMA, or gating. In the exploratory mixed population, PERG is a dynamically switching ZD

[2]Source code will be released when published.



(a) Population composition (b) Group cooperation

Figure 4: Five-strategy Grid competition over 31 runs from equal 20% shares: (a) population composition; (b) cooperation rate. Shading shows pointwise 95% bootstrap intervals.

baseline (Huang et al., 2026), MTBR is its supplied table-based reciprocal baseline, and AllD and AllC provide unconditional reference behaviors.

5.2 Results

Overall performance against ablations and fixed ZD. Table 1 confirms that graph context significantly boosts population share, cooperation, and social welfare over *w/o GNN* (all $p_H < 0.001$), isolating the isolated effect of graph-informed representations. GraphZD similarly outpaces fixed endpoints (Fixed ZD-GEN/EX), highlighting the reliability of continuous adaptation. Seed-level distributions appear in Appendix Figure 7.

Evolutionary population dynamics and social welfare. Figure 3 shows that the main differences arise during invasion rather than only at the terminal state. GraphZD increases its population share rapidly across the three canonical topologies and the moderate SBM, while Fixed ZD-EX remains near its initial or lower prevalence and Fixed ZD-GEN spreads less consistently on heterogeneous networks. The *w/o GNN* variant can eventually approach a high final share, but reaches majority more slowly and less reliably. The social-welfare trajectories show the same ordering during the transition, linking early carrier survival to later cooperative consolidation. Appendix B.1 gives the corresponding network-level summaries.

Competition in mixed strategy population. Figure 4 extends the controlled invasion setting to a heterogeneous population. GraphZD, PERG (Huang et al., 2026), MTBR, AllD, and AllC start at equal 20% shares on a periodic Grid in 31 runs of 5,000 generations. Final-500 cooperation averages 96.7%; mean final shares are 32.8% GraphZD, 27.3% MTBR, 24.3% PERG, 15.6% AllC, and 0.03% AllD. Thus, GraphZD remains competitive without eliminating other cooperative strategies, while AllD nearly disappears. Because only one topology is tested and five seeds were previously screened, this comparison remains exploratory; Appendix A gives the complete protocol.

Discussion. GraphZD achieves full fixation across topologies, outperforming non-GNN and fixed baselines in final share, mid-run cooperation, and social welfare. This stems from a two-stage dynamic: network context aids early survival against AllD, enabling carriers to build high-welfare cooperative clusters that dominate mixed populations. While demonstrating these dynamic benefits, the results do not isolate individual update mechanics or prove universal strategy dominance.

6 Conclusion

GraphZD uses graph context to adapt enforceable payoff relations while preserving ZD feasibility. Evaluated across 90 runs per method, GraphZD reaches fixation in every trial. Relative to *w/o GNN*, it raises mean population share and mid-run cooperation by 2.9 and 11.8 percentage points, respectively, and reaches majority 66 generations earlier on average. Relative to Fixed ZD-GEN, the corresponding gains are 17.8 and 21.6 percentage points, together with a 0.842 increase in mid-run social welfare. Exploratory mixed-population experiments further reach 96.7% late-stage cooperation while retaining multiple cooperative strategies. These results support a two-stage account in which graph-aware control improves early survival under defector pressure, enabling later cooperative consolidation and higher social welfare.

AI USE STATEMENT

In this work, we used generative AI tools for assist with translation. We have not used generative AI tools for generate synthetic data sets, help develop theoretical models or conceptual frameworks, formulate mathematical claims, provide critical ingredients for proving mathematical claims, assist in the writing of proofs, propose or refine hypotheses, design or provide feedback on research methodology or experiments, implement methods, clean and reformat dataset, and support qualitative and thematic data analysis are not applicable to this work. Additionally, we used generative AI tools for editing software code. LLM-edited code was verified and tested for correctness, tested under the reported experimental configurations, and checked against the saved logs and numerical outputs. No experimental result was reported without executing the corresponding code. We have reviewed all AI-assisted work. We take responsibility for the final content of this work, including text, claims or artifacts produced with the aid of generative AI.

ETHICS STATEMENT

This work studies cooperation in simulated evolutionary games on synthetic static raphs. It does not involve human participants, personal or sensitive data, eal-world interventions, or the release of a new dataset. The reported outcomes escribe artificial agents under specified payoff and replacement rules and hould not be interpreted as prescriptions for human behavior or as evidence bout real social groups. The method can adapt payoff relations in a controlled imulation, but we do not claim that it is suitable for deployment in settings where automated incentives could affect people. We report negative and xploratory findings separately from the confirmatory results and state the cope of the empirical and theoretical claims to reduce the risk of vergeneralization.

REPRODUCIBILITY STATEMENT

The problem setting, payoff matrix, memory convention, fitness definition, and death–birth update are specified in Section 3. The GraphZD mapping and its two-stage procedure are described in Section 4 and the accompanying algorithms. Section 5.1 reports the main evaluation protocol. Appendix A documents offline encoder training, software and hardware environments, hyperparameters, metric definitions, statistical tests, and artifact tracking; Appendix A.3 gives the exact features and feedback signals. The feasibility proof and its numerical audit appear in Appendices D.1 and D.2. Per-topology results, seed-level visualizations, and the receptive-field, population-scale, SBM, and trainable-encoder extensions are reported in Appendices B.1–C.4. Completed runs are cached independently by method, topology, and seed, and the source code will be released upon publication.

REFERENCES

Guillermo Abramson and Marcelo Kuperman. Social games in a social network. *Physical Review E*, 63(3):030901, 2001.

Christoph Adami and Arend Hintze. Evolutionary instability of zero-determinant strategies demonstrates that winning is not everything. *Nature Communications*, 4:2193, 2013.

Juan P. Amboage, Ernst Roell, Patrick Schnider, and Bastian Rieck. LEAP: Local ECT-based learnable positional encodings for graphs. In *International Conference on Learning Representations*, 2026.

Robert Axelrod and William D Hamilton. The evolution of cooperation. *Science*, 211(4489): 1390–1396, 1981.

Federico Barbero, Ameya Velingker, Amin Saberi, Michael Bronstein, and Francesco Di Giovanni. Locality-aware graph rewiring in GNNs. In *International Conference on Learning Representations*, 2024.

Daizhan Cheng and Changxi Li. Design of zero-determinant strategies and its application to networked repeated games. *Science China Information Sciences*, 67(10):202201, 2024. doi: 10.1007/s11432-023-4022-6.

Jeongwhan Choi, Jongwoo Kim, Woosung Kang, and Noseong Park. Learning posterior predictive distributions for node classification from synthetic graph priors. In *International Conference on Learning Representations*, 2026.

Rohan Deb, Yikun Ban, Shiliang Zuo, Jingrui He, and Arindam Banerjee. Contextual bandits with online neural regression. In *International Conference on Learning Representations*, 2024.

Feng Fu, Te Wu, and Long Wang. Partner switching stabilizes cooperation in coevolutionary prisoner's dilemma. *Physical Review E*, 79(3):036101, 2009. doi: 10.1103/PhysRevE.79.036101.

Christian Hilbe, Martin A. Nowak, and Karl Sigmund. Evolution of extortion in iterated prisoner's dilemma games. *Proceedings of the National Academy of Sciences*, 110(17):6913–6918, 2013.

Huaihe Huang, Ye Ye, Wei Bao, Yue Zhang, and Nenggang Xie. An evolutionary game analysis of a dynamically switching zero-determinant strategy. *Applied Mathematics and Computation*, 515: 129862, 2026. doi: 10.1016/j.amc.2025.129862.

Kai Kang, Jinyan Tian, and Boyu Zhang. Cooperation and control in asymmetric repeated games. *Applied Mathematics and Computation*, 470:128589, 2024. doi: 10.1016/j.amc.2024.128589.

Parnian Kassraie, Andreas Krause, and Ilija Bogunovic. Graph neural network bandits. In *Advances in Neural Information Processing Systems (NeurIPS)*, volume 35, 2022.

Tor Lattimore and Csaba Szepesvári. *Bandit Algorithms*. Cambridge University Press, Cambridge, UK, 2020.

Lihong Li, Wei Chu, John Langford, and Robert E. Schapire. A contextual-bandit approach to personalized news article recommendation. In *Proceedings of the 19th International Conference on World Wide Web*, pp. 661–670. ACM, 2010.

Shouheng Li, Floris Geerts, Dongwoo Kim, and Qing Wang. Towards bridging generalization and expressivity of graph neural networks. In *International Conference on Learning Representations*, 2025.

Alex McAvoy and Christoph Hauert. Structural symmetry in evolutionary games. *Journal of The Royal Society Interface*, 12(111):20150420, 2015. doi: 10.1098/rsif.2015.0420.

Yao Meng, Alex McAvoy, and Aming Li. Promoting collective cooperation through temporal interactions. *Proceedings of the National Academy of Sciences*, 122(26):e2509575122, 2025. doi: 10.1073/pnas.2509575122.

Patrick Alfred Pierce Moran. Random processes in genetics. *Mathematical Proceedings of the Cambridge Philosophical Society*, 54(1):60–71, 1958.

Martin A. Nowak. *Evolutionary Dynamics: Exploring the Equations of Life*. Harvard University Press, Cambridge, MA, 2006.

Martin A. Nowak and Robert M. May. Evolutionary games and spatial chaos. *Nature*, 359(6398): 826–829, 1992.

Hisashi Ohtsuki, Christoph Hauert, Erez Lieberman, and Martin A. Nowak. A simple rule for the evolution of cooperation on graphs and social networks. *Nature*, 441(7092):502–505, 2006.

Jorge M. Pacheco, Arne Traulsen, and Martin A. Nowak. Coevolution of strategy and structure in complex networks. *Physical Review Letters*, 97(25):258103, 2006.

William H. Press and Freeman J. Dyson. Iterated prisoner's dilemma contains strategies that dominate any evolutionary opponent. *Proceedings of the National Academy of Sciences*, 109(26):10409–10413, 2012.

Yunzhe Qi, Yikun Ban, and Jingrui He. Graph neural bandits. In *Proceedings of the 29th ACM SIGKDD Conference on Knowledge Discovery and Data Mining*, pp. 1920–1931, 2023. doi: 10.1145/3580305.3599371.

Chendi Qian, Andrei Manolache, Kareem Ahmed, Zhe Zeng, Guy Van den Broeck, Mathias Niepert, and Christopher Morris. Probabilistically rewired message-passing neural networks. In *International Conference on Learning Representations*, 2024.

Francisco C. Santos and Jorge M. Pacheco. Scale-free networks provide a unifying framework for the emergence of cooperation. *Physical Review Letters*, 95(9):098104, 2005.

Petar Veličković, Guillem Cucurull, Arantxa Casanova, Adriana Romero, Pietro Liò, and Yoshua Bengio. Graph attention networks. In *International Conference on Learning Representations (ICLR)*, 2018.

Duncan J. Watts and Steven H. Strogatz. Collective dynamics of 'small-world' networks. *Nature*, 393(6684):440–442, 1998.

Zonghan Wu, Shirui Pan, Fengwen Chen, Guodong Long, Chengqi Zhang, and Philip S. Yu. A comprehensive survey on graph neural networks. *IEEE Transactions on Neural Networks and Learning Systems*, 32(1):4–24, 2020.

Zhipeng Zhang, Xiaotong Jiang, and Chengyi Xia. Zero-determinant strategy of finite games with implementation errors and its application into group decision-making. *Applied Mathematics and Computation*, 486:129055, 2025. doi: 10.1016/j.amc.2024.129055.

A Reproducibility and Implementation

A.1 Frequently Used Notation

Table 2: Frequently used notation

Symbol	Meaning
$\mathcal{G} = (\mathcal{V}, \mathcal{E})$	Fixed undirected interaction and replacement graph.
$N, \mathcal{N}(i), d_i$	Population size, neighbors of node i, and node degree $\lvert\mathcal{N}(i)\rvert$.
C, D	Cooperation and defection actions.
R, S, T, P	Reward, sucker's, temptation, and punishment payoffs; $(R, S, T, P) = (3, 0, 5, 1)$.
g, G, T_{IPD}	Current generation, total generations, and IPD rounds per edge and generation.
$\pi_i^{(g,t)}(j), \bar{\pi}_i(g)$	Stage payoff against neighbor j and mean interaction payoff of the player at node i.
$F_i(g)$	Fitness of the player at node i in generation g.
$\mathbf{p}_i$	Memory-one cooperation probabilities $(p_{i,CC}, p_{i,CD}, p_{i,DC}, p_{i,DD})$.
$\mathbf{f}_i, \mathbf{h}_i, \mathbf{x}_i$	Node features, GNN embedding, and contextual-controller input.
$\omega^\star$	Pretrained encoder parameters frozen in the confirmatory experiments.
s_g	Fraction of nodes occupied by GraphZD players in generation g.
$z_i, \widetilde{z}_i$	Executed continuous control and its pre-gating, pre-EMA proposal, both in $[-1, 1]$.
$\mathcal{Z}$	Nine control anchors $\{-1, -0.75, \ldots, 1\}$.
$\boldsymbol{\psi}(\mathbf{x}, z)$	Joint context–control feature $[\mathbf{x}, z\mathbf{x}, z^2\mathbf{x}]$.
$\mathbf{A}, \mathbf{b}_q, \widehat{\boldsymbol{\theta}}_q$	Regularized design matrix, reward vector, and ridge estimate for feedback q.
w_i, A_i, ℓ_i	Local welfare, fitness advantage over AllD, and post-replacement neighborhood retention.
α, τ, η, H	UCB coefficient, softmax temperature, EMA rate, and controller horizon.
$\chi(z), \ell(z), \mathbf{d}(z)$	ZD payoff factor, payoff baseline, and Press–Dyson direction induced by control z.
$\mathbf{o}, \phi_{\max}(z), \phi(z)$	Memory-one offset, largest common feasible scale, and executed common scale.

A.2 Offline Encoder Training

The frozen encoder is learned from a synthetic corpus rather than from any confirmatory trajectory. We generate $N = 100$ Erdős–Rényi, random-regular, and stochastic-block-model graphs using 40 training seeds and 10 disjoint validation seeds, with two independently sampled behavioral snapshots per graph. This gives 240 training and 60 validation snapshots. Grid, scale-free, and small-world graphs are absent from both splits and are used only for downstream evaluation. Each snapshot assigns one of four latent behavioral profiles to the player at every node, perturbs each player's cooperation propensity with Gaussian noise, samples one action per player, and computes Prisoner's Dilemma payoffs on the graph. The four encoder inputs are historical cooperation, mean payoff divided by five, degree divided by the graph maximum, and the most-recent-defection indicator.

Pretraining uses the two-layer GAT with a temporary sigmoid head $\widehat{y}_i$. For sampled action y_i, min–max-normalized payoff fitness $\widetilde{F}_i$, synthetic carrier indicator m_i, and carrier fraction $r = N^{-1}\sum_i m_i$, the loss is

$$\mathcal{L}_{\mathrm{off}} = (1 - r)\frac{1}{N}\sum_i \widetilde{F}_i(\widehat{y}_i - y_i)^2 + r\frac{\sum_i m_i(\widehat{y}_i - 1)^2}{\sum_i m_i + 10^{-6}}.$$

Algorithm 2 summarizes the separation between offline representation learning and confirmatory evaluation. We optimize with Adam for 120 epochs at learning rate 10^{-3} and retain the checkpoint with the lowest validation loss. The temporary head is then discarded. Confirmatory inference reuses $\omega^\star$ but restricts the second GAT layer to self-loops, as specified in Section 4; hence only the first layer communicates across edges. The synthetic objective supplies a stable initialization and is not used as evidence of evolutionary performance. Reuse across unseen graph families is therefore treated as

Algorithm 2: Offline Training of the One-Hop Graph Encoder

Input : Synthetic graph families; disjoint training and validation seeds; epochs E
Output : Frozen encoder parameters $\omega^\star$

```
Generate training and validation graphs with two behavioral snapshots per graph;
Construct node features, sample player actions, and compute normalized payoffs and synthetic
  carrier labels;
Initialize the two-layer GAT encoder f_ω and a temporary prediction head;
for e = 1 to E do
    Update ω and the temporary head by minimizing L_off on the training split;
    Retain the parameters with the lowest validation loss;
Discard the temporary head and freeze the selected encoder ω*;
```

an empirical transfer test rather than a generalization guarantee; recent theory likewise relates GNN generalization to the structural variation represented by its embeddings Li et al. (2025).

A.3 Features and Feedback Signals

For node i, the encoder input and normalized controller context used in Section 4 are

$$\mathbf{f}_i(g) = [c_i(g),\ \bar{r}_i(g)/5,\ d_i/d_{\max},\ b_i(g-1)]\,,$$
$$\mathbf{x}_i(g) = \left[1,\ s_g,\ g/(G-1),\ \frac{\mathbf{h}_i(g)}{\max(\|\mathbf{h}_i(g)\|_2, \epsilon)}\right] \in \mathbb{R}^{19}.$$

Here c_i is historical cooperation (zero for non-carriers), $\bar{r}_i$ is recent mean payoff with a neutral pre-observation value, d_i is degree, $b_i(g-1)$ marks the most recently observed defection, and s_g is the current GraphZD population share. The three controller feedback signals are

$$w_i = \mathrm{clip}\left((J_i - 2)/4, 0, 1\right), \qquad A_i = \tanh\left(F_i - \bar{F}_{D,i}\right),$$
$$\ell_i = \frac{m_i^+ + \sum_{j\in\mathcal{N}(i)} m_j^+}{1 + d_i},$$

where J_i is local publicsocial welfare, $\bar{F}_{D,i}$ is local mean AllD fitness with global backoff, and m_j^+ indicates post-replacement carrier occupancy. The last quantity is post-replacement feedback rather than an encoder input.

A.4 Experimental Hardware and Software

The original experimental batches ran on a laptop workstation with a 13th-generation Intel Core i9-13900H CPU (14 physical cores and 20 logical processors), 32 GB of system memory, and one NVIDIA GeForce RTX 4060 Laptop GPU with 8,188 MiB of device memory. The host operating system was 64-bit Windows 11 Pro (build 26200), and the experiments ran inside Windows Subsystem for Linux 2 (WSL2) with Ubuntu 24.04.1 LTS. Each run used a single GPU; methods within a network–seed pair were executed serially, and completed runs were cached independently. These batches used no distributed execution. The Fixed ZD-EX/ZD-GEN extension ran on a Linux server with two Intel Xeon Silver 4210R processors, 125 GiB of memory, and two NVIDIA GeForce RTX 4090 GPUs with 24,564 MiB. The five-strategy experiment in Section 5.2 also ran on a separate remote server; the laptop specification does not describe these extension batches.

A.5 Metrics and Statistical Inference

For the strategy population share s_g, mean population share is the normalized occupancy AUC $G^{-1}\sum_{g=0}^{G-1} s_g$. Cooperation is the fraction of cooperative actions, and population social welfare is the mean two-player payoff sum over active interactions. The mid-run window spans generations 1,000–3,999. Fixation is the first generation with $s_g = 1$ and is right-censored at generation 5,000. Inference uses paired bootstrap intervals, paired t tests, Wilcoxon tests, Cohen's d_z, and Holm correction.

Table 3: Frozen GraphZD hyperparameters.

Parameter	Value	Parameter	Value
GNN mode	one-hop, frozen	Controller horizon H	25
Context dimension	19	Joint dimension D	57
UCB coefficient α	0.25	Softmax temperature τ	0.10
EMA η	0.20	ZD scale ratio	0.95
$\chi_{\max}$	3.0	Emergency share	0.15
Bandit discount	1.0	Persistent edge memory	false

Table 4: Confirmatory results (mean $\pm$ standard deviation over 30 seeds).

Network	Method	Mean population share	Mid cooperation	Mid social welfare
Grid	GraphZD 1-hop	0.867 ± 0.051	0.933 ± 0.059	5.743 ± 0.227
	w/o GNN	0.843 ± 0.089	0.773 ± 0.238	5.125 ± 0.933
	Fixed ZD-EX	0.307 ± 0.241	0.026 ± 0.023	2.129 ± 0.115
	Fixed ZD-GEN	0.751 ± 0.166	0.788 ± 0.198	5.185 ± 0.786
Scale-free	GraphZD 1-hop	0.860 ± 0.071	0.941 ± 0.090	5.773 ± 0.350
	w/o GNN	0.821 ± 0.099	0.813 ± 0.212	5.291 ± 0.815
	Fixed ZD-EX	0.201 ± 0.232	0.016 ± 0.023	2.082 ± 0.112
	Fixed ZD-GEN	0.646 ± 0.301	0.696 ± 0.357	4.812 ± 1.426
Small-world	GraphZD 1-hop	0.873 ± 0.038	0.947 ± 0.045	5.799 ± 0.170
	w/o GNN	0.848 ± 0.098	0.882 ± 0.178	5.545 ± 0.700
	Fixed ZD-EX	0.344 ± 0.255	0.030 ± 0.026	2.148 ± 0.127
	Fixed ZD-GEN	0.669 ± 0.264	0.690 ± 0.312	4.793 ± 1.247

A.6 Five-Strategy Protocol

GraphZD, PERG, MTBR, AllD, and AllC each initially occupy 20 of the $N = 100$ Grid nodes. The payoff matrix and death–birth update match the confirmatory experiment, and each run uses 20 IPD rounds per edge and generation. MTBR uses its supplied table-based reciprocal rule with initial cooperation probability 0.5. The cohort contains 31 seeds, five of which were previously screened. Figure 4 uses pointwise 95% percentile bootstrap intervals from 2,000 whole-seed resamples, not simultaneous confidence bands.

A.7 Frozen Configuration and Artifact Tracking

The frozen manifest records hashes for the encoder, controller, engine, topology builder, formal runner, configuration, and regression tests. Results are cached per method, network, and seed; aggregation preserves completed runs.

B Results and Diagnostics

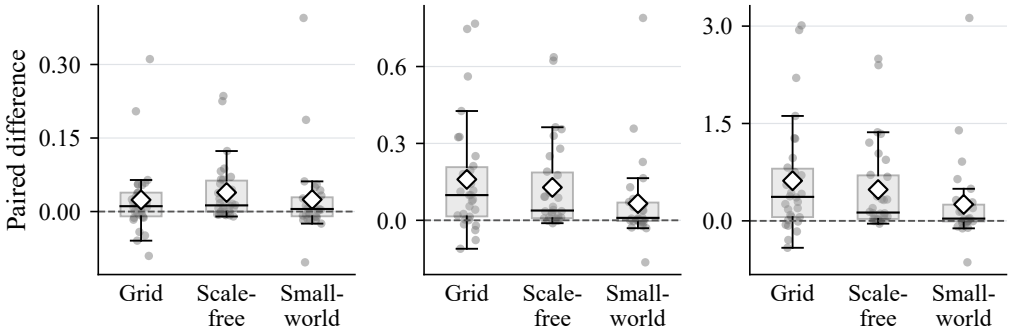
B.1 Detailed Results

The main text reports the pooled paired estimates and complete trajectories. Table 4 gives the per-network values and trajectory-level interpretation. Appendix B.2 provides supplementary seed-level plots for the same comparisons.

Trajectory-level outcomes distinguish the controllers more clearly than terminal share. GraphZD reaches final share 1.0 in all 90 runs, while Fixed ZD-GEN averages final shares of 0.964, 0.810, and 0.895 on Grid, scale-free, and Small-world, respectively; Fixed ZD-EX remains far lower. During invasion, GraphZD improves Grid mid cooperation by 0.160 and mid payoff by 0.618 over *w/o GNN*, and by 0.145 and 0.558 over Fixed ZD-GEN. The corresponding GraphZD–Fixed ZD-GEN gains are 0.245 and 0.961 on scale-free and 0.257 and 1.006 on Small-world. Thus generosity is a useful endpoint but does not reproduce the robustness of structure-aware adaptation.

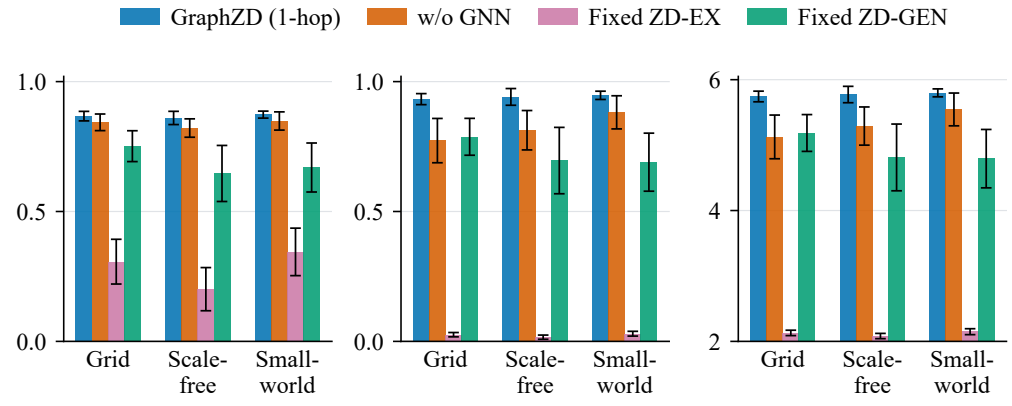
Mechanistic interpretation. Together, the ablation and fixed-endpoint comparisons support a two-stage account. Early in a run, minority carriers must avoid a persistent local fitness deficit near defectors. Once they survive and form larger connected groups, carrier–carrier interactions become more frequent and generosity-oriented control can increase cooperation and welfare. Graph context is relevant because the same neighborhood defines both repeated-game partners and replacement competitors, allowing the controller to distinguish exposed boundary nodes from locally supported carriers. The mixed-population result complements this invasion setting: high cooperation does not require GraphZD to eliminate every cooperative alternative. These trajectories are consistent with the proposed mechanism, but do not isolate the causal effect of every control update.

B.2 Additional Visualizations



(a) Mean population share (b) Cooperation (c) Joint payoff

Figure 5: Paired one-hop GraphZD minus *w/o GNN* differences, with seed points, median/interquartile boxes, and mean diamonds.



(a) Mean population share (b) Cooperation (c) Joint payoff

Figure 6: Formal summary of GraphZD performance across different topologies.

B.3 Linear-Model and Excitation Diagnostics

The diagnostics assess whether useful reward prediction is accompanied by the excitation required for parameter consistency. We log design spectra, prequential error, and leave-one-seed-out prediction.

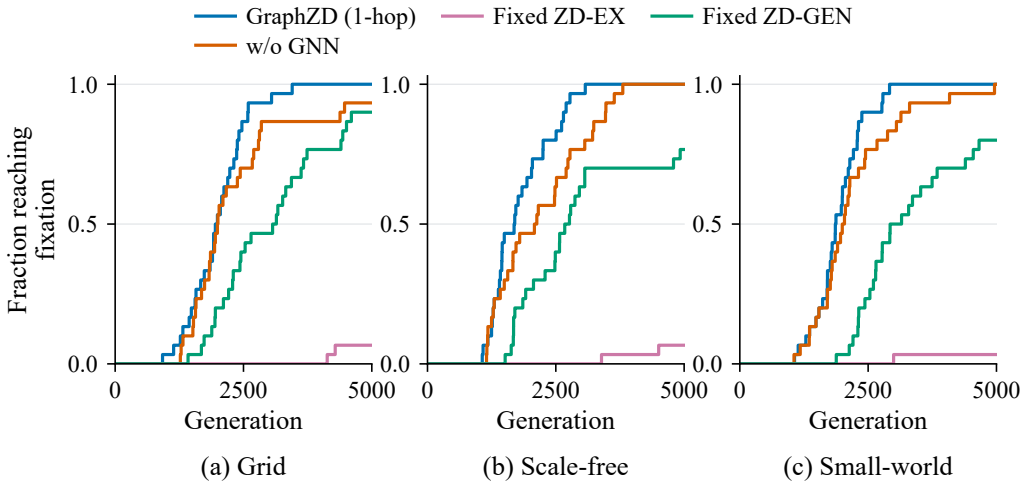


Figure 7: Cumulative fixation timing; runs not fixed by generation 5,000 remain right-censored. Terminal share is nearly saturated, but fixation timing shows that the main separation occurs during invasion rather than only at the endpoint.

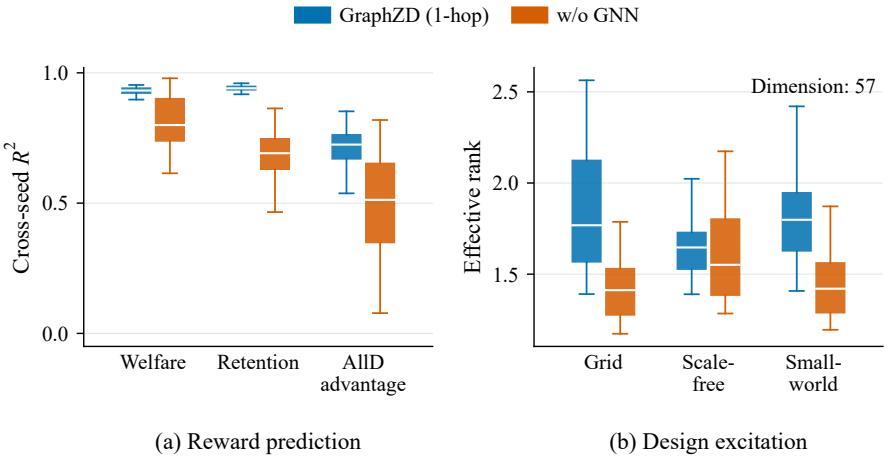


Figure 8: Linear-controller diagnostics over the formal batch. (a) Leave-one-seed-out reward-prediction R^2, pooled over three topologies but separated by reward and controller; "Retention" denotes post-replacement neighborhood feedback. (b) Effective rank by topology after deduplicating reward-repeated design diagnostics. The nominal feature dimension is 57.

Pooling the two adaptive variants within each topology, mean effective rank ranges from 1.63 to 1.65 despite $D = 57$; normalized minimum eigenvalues are near zero, and some condition numbers are infinite. With the same pooling, topology-specific mean cross-seed R^2 ranges from approximately 0.83 to 0.93 for welfare, 0.78 to 0.84 for post-replacement neighborhood feedback, and 0.57 to 0.64 for AllD advantage. Figure 8 additionally separates the controllers. The results support prediction on the visited subspace without establishing full-dimensional persistent excitation, leaving Theorem 3 a conditional module-level statement.

The trajectories offer a plausible explanation for the low rank. Population share and normalized generation vary smoothly, EMA restricts control variation, and many runs converge toward a common high-cooperation state. The nominal 57-dimensional feature space is therefore sampled unevenly. AllD-advantage prediction may also become harder late in successful invasions, when few defectors remain for local comparison. More informative probing could address this limitation, but would need to account for the invading minority's risk of extinction.

Table 5: Receptive-field ablation over five paired seeds per topology. Mean GraphZD population share is averaged over all 5,000 generations.

Network	Variant	Mean population share	Mid social welfare	Mean fixation generation
Overall	Full 2-hop	0.851	5.667	2002.1
	1-hop GNN	0.864	5.760	1832.0
	w/o GNN	0.849	5.588	2053.3
Grid	Full 2-hop	0.828	5.511	2254.0
	1-hop GNN	0.832	5.609	2133.4
Scale-free	Full 2-hop	0.871	5.830	1738.8
	1-hop GNN	0.875	5.806	1691.6
Small-world	Full 2-hop	0.853	5.659	2013.6
	1-hop GNN	0.886	5.865	1671.0

C Exploratory Extensions

C.1 Receptive-Field Analysis

We tested whether extending the frozen encoder from one-hop to two-hop message passing improves evolutionary control. In the one-hop method used in the main paper, the first GAT layer aggregates immediate neighbors and the second layer is restricted to self-loops. The "Full 2-hop" variant allows both GAT layers to aggregate neighbors. The continuous controller, feasible MS-ZD mapping, death–birth process, initial 20:80 GraphZD–AllD mixture, strong selection, and all other settings are unchanged. Runs last 5000 generations with 20 IPD rounds per edge. Each method uses the same five paired seeds on each of Grid, scale-free, and small-world networks (15 runs per method). This experiment therefore isolates receptive-field depth, but its five-seed-per-network sample is exploratory rather than confirmatory.

Extending the receptive field does not improve pooled performance in this pilot, with the clearest disadvantage on small-world. Across the 15 paired network–seed observations, Full 2-hop is lower than 1-hop by 0.0137 in mean GraphZD population share (paired-bootstrap 95% CI $[-0.0265, -0.0021]$) and 0.0934 in mid-run social welfare (95% CI $[-0.1744, -0.0100]$). Mean fixation is 170 generations later, although its interval crosses zero. On small-world, the differences are -0.0334 in mean population share (95% CI $[-0.0543, -0.0120]$), -0.2058 in mid payoff (95% CI $[-0.3297, -0.0771]$), and 342.6 generations later fixation (95% CI $[100.4, 662.6]$). On scale-free, the population-share difference is small and two-hop mid payoff is slightly higher, with both intervals crossing zero. On Grid, the extra hop does not improve population share and lowers mid payoff by 0.0988 (95% CI $[-0.2042, -0.0109]$), again favoring one-hop in this pilot.

The ordering favors the frozen one-hop encoder under the present settings. A possible explanation is its alignment with local death–birth competition: a second aggregation step may mix boundary and interior signals and weaken contrasts relevant to the next replacement event. This explanation remains a hypothesis, especially with only five seeds per topology. More generally, recent ICLR studies show that longer-range message passing can face locality and over-squashing trade-offs, and may require explicit rewiring rather than additional aggregation alone Barbero et al. (2024); Qian et al. (2024). Two-hop information may become more useful with a training target that predicts propagation, residual multi-scale features, or dynamic contacts; the current results do not establish a general advantage of shallower GNNs.

C.2 Population-Scale Receptive-Field Analysis

We test whether the value of structural context changes with population size using $N \in \{100, 400, 900\}$, an initial GraphZD–AllD ratio of 20:80, strong death–birth updating, and 20 IPD rounds per edge and generation. Each run lasts $20N$ replacement events, or 20 population sweeps: 2000, 8000, and 18000 generations, respectively. This aligns evolutionary time across sizes. We compare *w/o GNN*, the frozen one-hop encoder, and Full 2-hop on all three topologies with five paired seeds per cell, giving 135 runs. The scaling experiment is separate from the 5000-generation confirmatory protocol and is interpreted descriptively.

Table 6: Mean GraphZD population share in the population-scale experiment. Each entry averages five paired seeds over 20 population sweeps.

N	Network	*w/o GNN*	1-hop GNN	Full 2-hop
100	Grid	**0.647**	0.592	0.581
	Scale-free	0.645	**0.686**	0.678
	Small-world	0.609	**0.715**	0.631
400	Grid	0.553	**0.654**	0.645
	Scale-free	0.353	**0.755**	0.580
	Small-world	0.508	**0.688**	0.656
900	Grid	0.443	**0.644**	0.570
	Scale-free	0.317	**0.721**	0.562
	Small-world	0.341	0.652	**0.656**

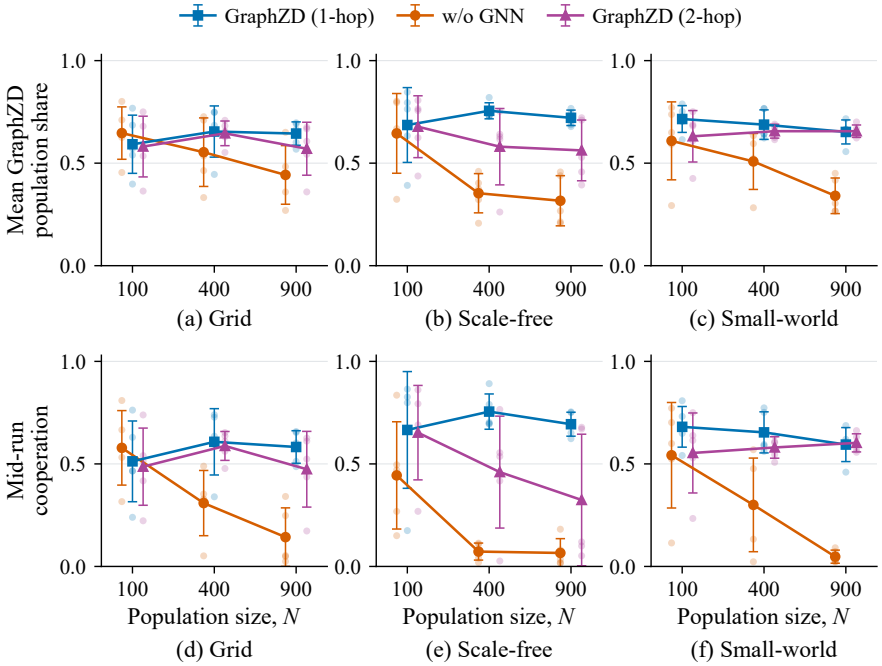


Figure 9: **Population size changes the value of graph context.** figure (a)–(c) show mean GraphZD population share and figure (d)–(f) show mid-run cooperation for Grid, scale-free, and small-world networks. GraphZD (2-hop) denotes the Full 2-hop variant. Large markers and lines report means over five paired seeds; error bars show one sample standard deviation, and translucent points show individual runs. Each experiment lasts 20 population sweeps, so the corresponding durations are 2,000, 8,000, and 18,000 generations for $N = 100, 400, 900$.

Figure 9 shows that the value of structural encoding increases with population size. Pooled across topologies, the one-hop mean-share advantage over *w/o GNN* rises from 0.031 at $N = 100$ to 0.227 at $N = 400$ and 0.306 at $N = 900$. The only negative comparison is Grid at $N = 100$ (-0.055); all $N = 400$ and $N = 900$ comparisons favor one-hop, and cooperation follows the same pattern. Full 2-hop does not improve monotonically: it remains below one-hop on Grid and scale-free and is only marginally higher on small-world at $N = 900$. Immediate-neighbor information may therefore align better with local death–birth competition, whereas a second aggregation step can mix interior and boundary signals. With five seeds per condition and time normalized by population sweeps, this is a scale-dependent mechanism hypothesis rather than an asymptotic law.

C.3 Degree-Matched SBM Extension

The SBM extension tests the main ordering while varying community strength at matched expected density. Four communities of 25 nodes use Strong, Moderate, and Weak $(p_{\text{in}}, p_{\text{out}})$ settings of $(0.150, 0.004)$, $(0.100, 0.020)$, and $(0.050, 0.036)$, respectively; each has expected mean degree 3.9, and disconnected samples are rejected. The remaining confirmatory protocol is unchanged, yielding 360 runs across three networks, four methods, and 30 paired seeds; Fixed ZD-Fair is excluded.

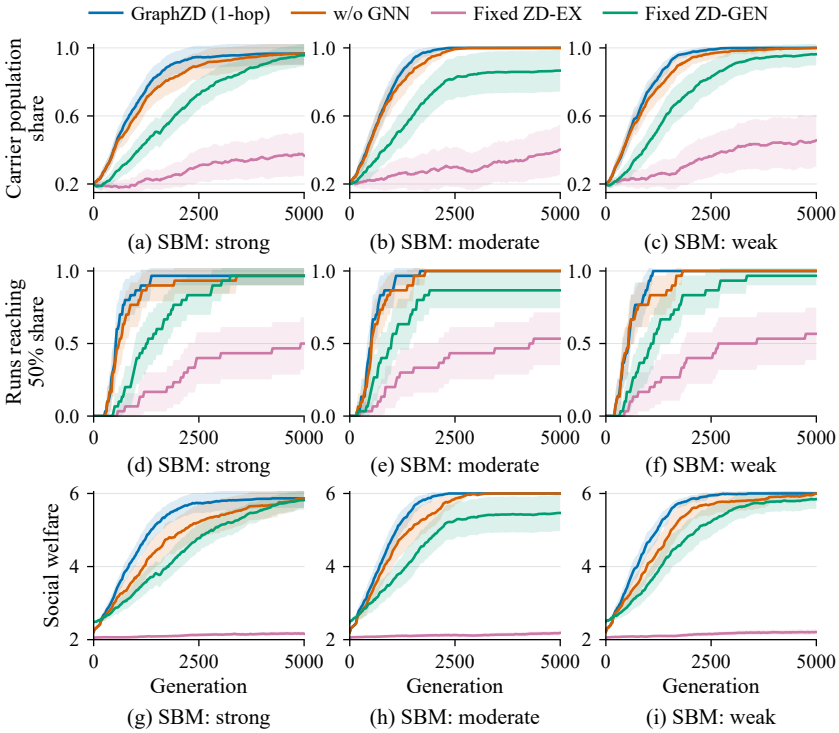


Figure 10: Complete four-method trajectories on degree-matched Strong, Moderate, and Weak SBMs. Rows show mean carrier population share, the cumulative fraction of runs that have reached 50% carrier population share, and social welfare (population mean social welfare). Curves aggregate 30 paired seeds, and shading gives 95% normal confidence intervals.

GraphZD retains the same ordering in all three networks (Figure 10). Across the 90 runs, its mean-share advantage over *w/o GNN* is 0.024 (95% CI $[0.010, 0.038]$); the corresponding differences are 0.090 $[0.052, 0.129]$ for mid-run cooperation and 0.345 $[0.195, 0.494]$ for mid-run social welfare. The differences remain positive within Strong, Moderate, and Weak SBMs. Relative to Fixed ZD-GEN, the stronger fixed endpoint, the pooled advantages are 0.166 in mean share, 0.216 in mid-run cooperation, and 0.839 in mid-run payoff; Fixed ZD-EX remains near defection-level cooperation. Strong modularity produces the widest intervals for both adaptive methods, consistent with sparse inter-community bridges increasing trajectory-to-trajectory variability. Moderate and Weak SBMs are more stable, but the non-monotone difference between them does not support a universal monotonic effect of modularity.

C.4 Trainable-GNN Variant

The trainable-GNN variant tests whether the frozen representation can be adapted online using observed controller feedback. It starts from the same one-hop GAT weights as the confirmatory method, uses an EMA target encoder for decisions, and minimizes summed Huber losses for welfare, AllD advantage, and post-replacement neighborhood retention. Algorithm 3 defines the update, with $\mathcal{C}_g$ the pre-replacement carrier set and $y_{i,\ell} = \ell_i$. Controller coefficients, selected controls, and realized feedback are held fixed during encoder updates, so gradients do not pass through action

sampling, payoff realization, or death–birth replacement. Training begins after generation 150, runs every 25 generations with clipped stochastic gradient descent (SGD), and stops when the GraphZD population share reaches 0.99. The base pilot uses learning rate 10^{-3}, gradient-norm limit 1, and target-EMA coefficient 0.01.

Freezing also separates the controller and representation time scales. If the GNN and contextual bandit are updated together, a change in the embedding moves the coordinate system in which the bandit's design matrix and reward vectors were accumulated. This statistical nonstationarity is most consequential early in a run: adaptive carriers are still a minority, death–birth outcomes are sparse, and the measured advantage over AllD has high variance. Small differences among controls may therefore be obscured or learned too late. Because death–birth trajectories are path dependent, a missed early survival advantage leaves fewer carriers and weaker cooperative clusters from which later population payoff can grow. This is a mechanism hypothesis motivating the frozen design, rather than a causal conclusion from the small pilot below.

Table 7: Exploratory trainable-GNN pilot, averaged over three networks and three paired seeds per network. The discounted variants use MAB discount $\gamma = 0.995$.

Variant	AUC	Final share	Mid payoff	Tail payoff
Frozen 1-hop	0.4196	0.7278	3.2418	4.5396
Frozen 1-hop + discounted MAB	0.4144	0.7000	3.1843	4.4079
Online 1-hop, no discount	0.4194	0.7267	3.2413	4.5371
Online 1-hop + discounted MAB	0.4097	0.6856	3.1716	4.3810

Algorithm 3: Exploratory online training of the one-hop GNN

Input : Pretrained encoder ω_0; warm-up g_0; update interval K; learning rate μ; gradient limit C; target-EMA rate ξ; stop share $s_{\max}$

Output : Online encoder ω and target encoder $\bar{\omega}$

Initialize $\omega \leftarrow \omega_0$ and $\bar{\omega} \leftarrow \omega_0$;

for $g = 0$ **to** $G - 1$ **do**

  Use $\bar{\omega}$ to construct one-hop node contexts and select player controls;

  Play the IPD interactions, perform one death–birth update, and observe $\mathbf{y}_i = (w_i, A_i, \ell_i)$;

  if $g \geq g_0$, $g \bmod K = 0$, *and the post-update share is below* $s_{\max}$ **then**

    Recompute $\boldsymbol{\psi}_i(\omega)$ from the pre-decision node features;

    $\mathcal{L}_g(\omega) \leftarrow \frac{1}{|\mathcal{C}_g|} \sum_{i \in \mathcal{C}_g} \sum_{q \in \{w, A, \ell\}} \mathrm{Huber}\Big(\boldsymbol{\psi}_i(\omega)^\top \widehat{\boldsymbol{\theta}}_q - y_{i,q}\Big)$;

    $\mathbf{g} \leftarrow \nabla_\omega \mathcal{L}_g(\omega)$; `// stop gradients through` z_i, $\widehat{\boldsymbol{\theta}}_q$, `and` $\mathbf{y}_i$

    $\widehat{\mathbf{g}} \leftarrow \mathbf{g} \min\{1, C/\max(\|\mathbf{g}\|_2, 10^{-12})\}$;

    $\omega \leftarrow \omega - \mu \widehat{\mathbf{g}}$;

    $\bar{\omega} \leftarrow (1 - \xi)\bar{\omega} + \xi\omega$;

  Update the contextual-controller statistics from $\mathbf{y}_i$;

We used a paired 2×2 pilot to separate GNN training from multi-armed bandit (MAB) discounting. Each setting contains the same three seeds on Grid, scale-free, and small-world networks, for nine paired runs per setting. Runs last 1,000 generations with 20 IPD rounds per edge. "AUC" below is the mean GraphZD population share across those generations. Because this protocol is shorter and much smaller than the 5,000-generation, 30-seed confirmatory batch, the results are descriptive and no significance claim is made.

Online encoder updates yield little practical improvement in this pilot, while MAB discounting accounts for the larger losses. Without discounting, conservative updates change paired mean AUC by -0.00021, with nine network–seed differences ranging from -0.00219 to 0.00038. On one scale-free seed, final share falls from 0.76 to 0.34 with discounting alone and 0.33 with discounting plus encoder updates. Stronger updates without discounting, using learning rate 0.01 and target-EMA coefficient 0.1, give AUC 0.4202 versus 0.4196 and final share 0.7300 versus 0.7278 for the paired frozen comparator, but mid-run and tail payoff differences of -0.0019 and -0.0144. The

optimization itself remains numerically stable; the concern is a moving representation coupled to sparse early feedback. Assessing that mechanism requires the main experiment's 5,000-generation, 30-seed protocol and an objective targeting propagation or future death–birth success.

C.5 Propagation-Aware Mechanism Extension

This extension tests whether explicit propagation feedback helps the controller respond to local replacement pressure. The 5-seed, 5,000-generation mechanism experiment adds an independent raw posterior, a graph/DB propagation residual, and a boundary gate. With only five paired seeds per network, it provides exploratory mechanism evidence.

Table 8: Propagation-aware mechanism results (mean over five seeds).

Network	Mean population share	Mid cooperation	Mid social welfare
Grid	0.837	0.795	5.225
scale-free	0.876	0.952	5.821
Small-world	0.863	0.884	5.559

The extension offers mixed evidence for propagation feedback, with benefits depending on topology and component choice. On scale-free, it performs approximately as well as the frozen one-hop method and exceeds its *w/o propagation head* ablation by 0.027 in mean population share, 0.055 in mid cooperation, and 0.219 in mid payoff. On Grid, removing the boundary gate raises mid cooperation from 0.795 to 0.894. These results motivate further mechanism tests while retaining the frozen one-hop controller as the main method.

D Theoretical and Numerical Audits

D.1 Proof of Feasibility and Direction Preservation

For every positive coordinate d_j, Equation (4) includes the upper bound $(1 - o_j)/d_j$; for every negative coordinate, it includes $-o_j/d_j$. A zero coordinate leaves $o_j + \phi d_j = o_j$ unchanged. Hence $0 \le o_j + \phi d_j \le 1$ for every j. Multiplying the common feasible scale by 0.95 preserves these inequalities and the common Press–Dyson direction, without component-wise clipping.

D.2 Probability Feasibility Audit

The numerical audit checks the implemented probability bounds against Proposition 1. Evaluating Equation (5) at 20,001 uniformly spaced controls in $[-1, 1]$ with the released JAX implementation gave finite probabilities with minimum 0, maximum 1, and no elements outside $[0, 1]$. This implementation check complements the proof.

D.3 Conditional Linear-Bandit Analysis

This analysis specifies what a linear-UCB surrogate can guarantee under a conditional reward model. Because the evolutionary system changes its own contexts and opponent population, the statement concerns the linear prediction layer rather than the full evolutionary process.

Assumption 2 (Conditional linear model)**.** *For the observations used by the controller, $\|\boldsymbol{\psi}_t\|_2 \le L$, $\|\boldsymbol{\theta}^*\|_2 \le S$, and $r_t = \boldsymbol{\psi}_t^\top \boldsymbol{\theta}^* + \epsilon_t$, where ϵ_t is conditionally R-sub-Gaussian. The reference policy chooses the maximum-UCB candidate from the finite set $\mathcal{Z}$ without EMA smoothing.*

Theorem 3 (Conditional surrogate regret)**.** *Under Assumption 2, regularized linear UCB on the $D = 57$ dimensional features in Equation* (1) *satisfies, with probability at least $1 - \delta$,*

$$\mathrm{Regret}(T) = O\left(\beta_T \sqrt{DT \log\left(1 + \frac{TL^2}{D\lambda}\right)}\right),$$

where $\beta_T = R\sqrt{D \log((1 + TL^2/\lambda)/\delta)} + \sqrt{\lambda} S$.

The theorem bounds regret for the unsmoothed surrogate against the best anchor on the realized contexts under the assumed linear reward model. It does not cover softmax averaging, identification probes, EMA, coexistence gating, endogenous death–birth contexts, fixation probability, or emergence of cooperation, nor does its comparator optimize the graph, update rule, or strategies outside the MS-ZD family. Low surrogate regret can therefore coexist with slow population growth, and successful fixation does not validate the linear assumptions. Parameter consistency additionally requires persistent excitation, for example $\lambda_{\min}(T^{-1}\sum_t \boldsymbol{\psi}_t\boldsymbol{\psi}_t^\top) > 0$; the diagnostics in Appendix B.3 do not support this full-dimensional condition on the observed trajectories.