NEURAL SOLVERS FOR MEAN-FIELD GAMES WITH EQUILIBRIUM CERTIFICATES

Anonymous authors
Paper under double-blind review

ABSTRACT

Large-scale multi-agent systems can be modeled by mean-field games (MFGs), in which individual decision-making and population evolution are coupled through an equilibrium between individual optimality, where the representative agent's policy is a best response to the population distribution, and population consistency, where the distribution is generated by the resulting policy. Solving such an equilibrium typically requires repeated forward–backward iterations to a fixed point for each new environment, objective, or population state. Neural solvers can reduce per-instance computation by learn a mapping from instances to solutions, but their predicted policies and populations need not form an equilibrium. We introduce Certified Equilibrium Neural Operator (CENO), a value-only neural MFG solver that couples solution construction with equilibrium certification. CENO predicts only candidate value functions. A Gibbs policy is derived from the predicted values and the population is then propagated through the policy-induced dynamics, ensuring population consistency by construction. To assess individual optimality, we derive an equilibrium certificate from the Bellman defect of the predicted values. The certificate upper-bounds the exploitability of the resulting policy-population pair, without computing a reference equilibrium or an additional best-response solve. The certificate also provides a label-free training objective, enabling the neural operator to learn across MFG instances without equilibrium solution labels. Under monotone population interactions, the certificate further bounds the error of the induced population trajectory relative to equilibrium, extending the guarantee from individual optimality to population-level accuracy. Experiments on a game-theoretic benchmark and three multi-agent population tasks evaluate solution quality and illustrate population-level coordination behaviors.

1 INTRODUCTION

Large-scale multi-agent systems arise in warehouse robotics(Barnhart et al., 2025), crowd evacuation(Toumi et al., 2024), traffic routing(Kang et al., 2020; Huang et al., 2021), and collective coordination(Wang et al., 2023). In such systems, each agent's decision is shaped by the state of the surrounding population, while the collective behavior of all agents in turn determines how the population evolves. As the number of agents grows, modeling the interactions among individuals and their impacts on the entire population becomes increasingly expensive. Mean-field games (MFGs) provide a scalable alternative(Lauriere et al., 2022; Dayanıklı & Lauriere, 2024): the population is described by a distribution, and a representative agent makes decisions against this distribution, turning large-scale interaction into a coupling between individual decisions and population evolution. Under this formulation, an MFG solution is a fixed point at which two processes agree at an equilibrium(Perolat et al., 2021; Han et al., 2024): the backward optimal decision of an agent against the current population state, and the forward evolution of the population under the resulting policy. In practice, however, every change in the environment layout, the task objective, or the current population state creates a new MFG instance. Classical MFG methods must solve this forward-backward iteration to convergence repeatedly for each new instance(Tang & Song, 2024), which becomes computationally costly in large systems that replan frequently.

To reduce the cost of repeated solving, recent work turns to neural methods for MFGs. Early approaches use networks to represent the value function, population distribution, or policy(Ruthotto

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.

et al., 2020; Lin et al., 2021), and solve each MFG instance by minimizing the residuals of the forward-backward equations(Assouli et al., 2025; Carmona & Laurière, 2022). While these methods alleviate the computational difficulty of high-dimensional state, a new instance still requires solving the forward-backward system again(Firoozi et al., 2025). More recent methods instead learn solution operators over a family of MFG instances, allowing a trained model to map a new instance directly to its solution in a single forward pass(Huang & Lai, 2025; Zeng et al., 2025; Thöni et al., 2026). Their accuracy is typically studied through residual, approximation, or generalization errors(Huang & Lai, 2024; Luo & Zheng, 2023), but these quantities do not directly measure whether the produced policy-population pair is close to an MFG equilibrium. A natural measure of equilibrium quality is exploitability(Perrin et al., 2020; Cui & Koeppl, 2021), which measures how much an individual agent can gain by deviating from the current policy while the population trajectory is fixed. However, computing it requires solving an additional best-response problem for each new instance. This brings back the per-instance computation that neural solvers are intended to avoid. We therefore ask: can a neural MFG solver produce a solution for a new instance without per-instance forward-backward iteration, while assessing its equilibrium quality from its own output alone, without the true equilibrium or an additional best-response solve?

We address this problem by treating the two requirements of an MFG equilibrium separately: we enforce forward consistency by construction and then measure the remaining backward optimality. Instead of predicting the policy and population distribution independently, our method, Certified Equilibrium Neural Operator (CENO), predicts only the value function of the entropy-regularized MFGs. A Gibbs policy is derived from the predicted value, and the population is generated by the policy-induced Markov dynamics(Cohen et al., 2024; Saldi, 2026). The resulting population trajectory is therefore always generated by its own policy, so forward consistency holds exactly. This leaves backward optimality as the only remaining equilibrium condition. We use the deviation of the predicted value function from the Bellman optimality relation to construct an equilibrium certificate, and prove that it upper-bounds the exploitability of the resulting policy-population pair, without the true equilibrium or an additional best-response solve. Because the certificate is computed entirely from the solver output and the known MFG model, it can be used both to assess each predicted solution and as a training objective without equilibrium labels. Under monotone interactions, we derive that this certificate also controls the error of the generated population trajectory, extending the guarantee from individual optimality to population evolution. This bound is further tightened to be an accumulated population-error bound under strong monotonicity.

Building upon these ideas, CENO produces a solution for a new instance in a single forward pass while providing a certificate of its equilibrium quality. In addition to the game-theoretic equilibrium benchmark, we compare CENO with other multi-agent methods on three multi-agent tasks covering different forms of population dynamics: spatial allocation through unequal-capacity routes, temporal adaptation to a time-varying bottleneck, and coordination between opposing populations in a shared space. The results show that, in addition to fast solving and good equilibrium quality, CENO produces population-level coordination behaviors that are not explicitly prescribed. Our main contributions are as follows:

- **Value-based neural solver for prediction and certification.** Instead of predicting the policy or population independently, we use predicted value function to construct a policy-population pair and to evaluate its Bellman defect. The policy-induced population evolution enforces forward consistency.
- **Computable objective with exploitability guarantee.** We derive a shift-invariant certificate that upper-bounds the exploitability and can be minimized without equilibrium labels or a backward best-response computation. This certificate also serves as training objective.
- **Population-level bounds from the same certificate.** Under monotone interactions, we bound the error of the induced population trajectory by the same certificate. Strong monotonicity additionally yields an accumulated population-error bound.

2 Related Work

Recent work on neural MFG solvers increasingly focuses on solutions that generalize across problem instances. The Master Policy(Perrin et al., 2022) conditions the policy on the population distribution, allowing one network to adapt across different initial populations. More recent solution-operator

approaches(Huang & Lai, 2024; Zeng et al., 2025) learn mappings from problem conditions to MFG solutions and can therefore produce solutions for new instances without additional solving. These methods mainly study solution accuracy through equation residuals, approximation errors, or generalization across instances. Researchers also utilize exploitability to evaluate equilibrium, as in fictitious play(Perrin et al., 2020), online mirror descent(Perolat et al., 2021), and entropy-regularized learning(Cui & Koeppl, 2021). Such evaluation requires computing a best response to the resulting population trajectory and is therefore typically performed post-hoc. In addition, another challenge is that these methods predict the policy and the population independently, which cannot guarantee that the policy is consistent with the population evolution. The trajectory-based solution operator(Huang & Lai, 2024) makes population evolution consistent through the push-forward of predicted trajectories, but cannot provide extra information to evaluate equilibrium. CENO instead predicts the value function and generates the policy and population from it, making the forward consistency exact while leaving the Bellman defect as an explicit remaining error.

3 PRELIMINARIES

3.1 FINITE-STATE MEAN-FIELD GAMES

We consider a finite-horizon entropy-regularized MFG with finite state space $\mathcal{X} = \{1, \dots, S\}$, time index $h = 0, \dots, H$, and a finite nonempty admissible action set $\mathcal{A}(x)$ at each state x. Let

$$\Pi = \{\pi = (\pi_0, \dots, \pi_{H-1}) : \pi_h(\cdot \mid x) \in \Delta(\mathcal{A}(x)), \forall h, x\} \tag{1}$$

denote the policy space. The population state at time h is a probability distribution

$$m_h \in \Delta(\mathcal{X}) := \left\{ m \in \mathbb{R}^S : m(x) \geq 0, \sum_x m(x) = 1 \right\}. \tag{2}$$

For each h, the population-independent transition kernel satisfies $P_h(\cdot \mid x, a) \in \Delta(\mathcal{X})$ for every admissible state-action pair (x, a). The stage cost is $c_h(x, a) + f_h(x, m_h)$, where c_h is the individual task cost and $f_h : \mathcal{X} \times \Delta(\mathcal{X}) \to \mathbb{R}$ is the population interaction cost, and $g : \mathcal{X} \to \mathbb{R}$ is the terminal cost. Given a fixed population trajectory $m = (m_0, \dots, m_H)$, a representative agent chooses a policy $\pi \in \Pi$ to minimize

$$\mathcal{J}_m(\pi) = \mathbb{E}_\pi \left[\sum_{h=0}^{H-1} (c_h(X_h, A_h) + f_h(X_h, m_h) + \tau \log \pi_h(A_h \mid X_h)) + g(X_H) \right], \tag{3}$$

where entropy temperature $\tau > 0$, $X_h \in \mathcal{X}$ is the random state of representative agent, $X_0 \sim m_0$, $A_h \sim \pi_h(\cdot \mid X_h)$, and $X_{h+1} \sim P_h(\cdot \mid X_h, A_h)$. For a fixed m, minimizing $\mathcal{J}_m$ gives the backward soft Bellman recursion and its Gibbs best-response policy:

$$Q_h^m(x, a) = c_h(x, a) + f_h(x, m_h) + \sum_{y \in \mathcal{X}} P_h(y \mid x, a) \phi_{h+1}^m(y), \tag{4}$$

$$\phi_h^m(x) = -\tau \log \sum_{a \in \mathcal{A}(x)} \exp\left(-\frac{Q_h^m(x, a)}{\tau} \right), \quad \phi_H^m = g, \tag{5}$$

$$\pi_h^m(a \mid x) = \frac{\exp(-Q_h^m(x, a)/\tau)}{\sum_{b \in \mathcal{A}(x)} \exp(-Q_h^m(x, b)/\tau)}. \tag{6}$$

Conversely, a policy π_h induces the Markov transition matrix and the population forward evolves as

$$m_{h+1} = m_h M_{\pi_h}, \quad M_{\pi_h}(x, y) = \sum_{a \in \mathcal{A}(x)} \pi_h(a \mid x) P_h(y \mid x, a). \tag{7}$$

Starting from m_0, these updates define the forward population trajectory induced by policy π, denoted by $\Gamma(\pi, m_0) = (m_0, \dots, m_H)$. Thus, a population trajectory determines an individual best response through the backward Bellman recursion, while a policy determines the population trajectory through the forward dynamics.

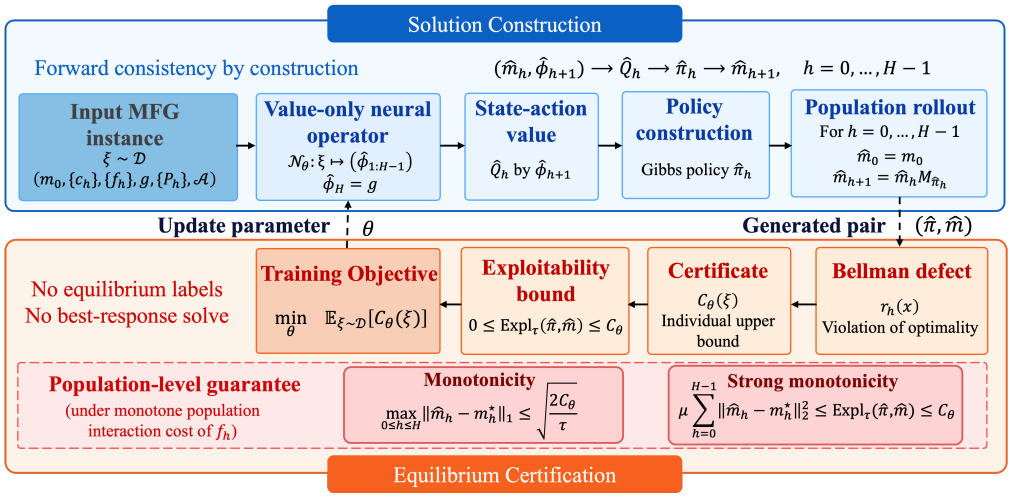


Figure 1: **Overview of CENO.** CENO predicts only continuation values, constructs a forward-consistent policy-population pair, and certifies its exploitability using the Bellman defect. Under monotone interactions, the certificate also yields population-level bounds.

3.2 Mean-Field Equilibrium and Exploitability

An MFG equilibrium is a policy-population pair $(\pi^\star, m^\star)$ satisfying two coupled conditions. First, forward consistency at the population level requires

$$m^\star_{h+1} = m^\star_h M_{\pi^\star_h}, \quad h = 0, \dots, H-1, \tag{8}$$

Second, backward optimality at the individual level requires

$$\pi^\star \in \arg\min_{\pi \in \Pi} \mathcal{J}_{m^\star}(\pi). \tag{9}$$

Hence an equilibrium is a fixed point between the population used to compute the best response and the population generated by that response. For a forward-consistent pair (π, m), we measure the remaining backward-optimality gap by exploitability:

$$\mathrm{Expl}_\tau(\pi, m) = \mathcal{J}_m(\pi) - \min_{\pi' \in \Pi} \mathcal{J}_m(\pi') \ge 0. \tag{10}$$

It is the reduction in cost available to the representative agent by switching to a best response while m is fixed. Therefore, for a forward-consistent pair, $\mathrm{Expl}_\tau(\pi, m) = 0$ if and only if backward optimality in Eq.(9) also holds, in which case (π, m) is an MFG equilibrium.

4 Methodology

4.1 Value-Only Solution Construction

For a fixed state space $\mathcal{X}$, admissible action sets $\mathcal{A}$, planning horizon H, and entropy temperature $\tau > 0$, we write an MFG instance as

$$\xi = (m_0, \{c_h\}_{h=0}^{H-1}, \{f_h\}_{h=0}^{H-1}, g, \{P_h\}_{h=0}^{H-1}), \quad \xi \sim \mathcal{D}. \tag{11}$$

An instance therefore specifies the initial population, the stage costs and population interactions over the horizon, the terminal cost, and the transition model. CENO uses a shared neural solver across this family and predicts only the continuation-value sequence:

$$\mathcal{N}_\theta : \xi \mapsto (\hat{\phi}_1, \dots, \hat{\phi}_{H-1}), \tag{12}$$

where $\hat{\phi}_h : \mathcal{X} \to \mathbb{R}$ with the terminal value fixed by the MFG terminal condition $\hat{\phi}_H = g$. The continuation values are predicted by the network rather than obtained by recursively solving the Bellman equations. Therefore, starting from the fixed initial population $\hat{m}_0 = m_0$, CENO constructs

the solution over $h = 0, \ldots, H-1$. At step h, substituting the current population $\hat{m}_h$ generated by the previous steps and the continuation value $\hat{\phi}_{h+1}$ predicted by the neural operator into Eq.(4) gives the state-action value $\hat{Q}_h(x,a)$ and corresponding Gibbs policy $\hat{\pi}_h$ from Eq.(6):

$$\hat{Q}_h(x,a) = c_h(x,a) + f_h(x,\hat{m}_h) + \sum_{y\in\mathcal{X}} P_h(y \mid x,a)\hat{\phi}_{h+1}(y), \tag{13}$$

$$\hat{\pi}_h(a \mid x) = \frac{\exp(-\hat{Q}_h(x,a)/\tau)}{\sum_{b\in\mathcal{A}(x)} \exp(-\hat{Q}_h(x,b)/\tau)}. \tag{14}$$

Instead of predicting the next population, CENO propagates the current population using the policy-induced transition matrix according Eq.(7)

$$\hat{m}_{h+1} = \hat{m}_h M_{\hat{\pi}_h} \quad M_{\hat{\pi}_h}(x,y) = \sum_{a\in\mathcal{A}(x)} \hat{\pi}_h(a \mid x) P_h(y \mid x,a). \tag{15}$$

Thus, with $\hat{m}_0 = m_0, \hat{\phi}_H = g$

$$(\hat{m}_h, \hat{\phi}_{h+1}) \longrightarrow \hat{Q}_h \longrightarrow \hat{\pi}_h \longrightarrow \hat{m}_{h+1}, \quad h = 0, \ldots, H-1. \tag{16}$$

We set $\hat{\phi}_H = g$ and do not predict $\hat{\phi}_0$, which does not affect the policy.

Exact forward consistency. By this construction,

$$\hat{m}_h \in \Delta(\mathcal{X}), \quad \hat{m}_{h+1} = \hat{m}_h M_{\hat{\pi}_h}, \quad h = 0, \ldots, H-1, \tag{17}$$

so the forward residual $R_h^{\text{fwd}} = \hat{m}_{h+1} - \hat{m}_h M_{\hat{\pi}_h}$ is identically zero.

The proof and corresponding properties are given in Appendix A.1. Forward consistency is structurally guaranteed and does not depend on the accuracy of the predicted continuation values. Even a poor value prediction may induce a non-optimal policy, but forward consistency of the policy-population pair still holds exactly. Backward optimality is therefore the only remaining equilibrium condition. Importantly, forward consistency gives the equilibrium deviation a clear meaning: since the population trajectory is generated by the same policy being evaluated, the gain from a unilateral deviation measures the remaining backward-optimality gap of this policy-population pair.

4.2 Equilibrium Certificate and Training

For the forward-consistent pair $(\hat{\pi}, \hat{m})$ generated by CENO, the remaining equilibrium condition is backward optimality under the fixed population trajectory $\hat{m}$: given the population trajectory $\hat{m}$ generated by the solver itself, how close is the current policy $\hat{\pi}$ to a best response? Exact exploitability can be computed by an additional backward Bellman recursion against this trajectory. Instead, CENO bounds exploitability using the predicted continuation values and the quantities already constructed during the rollout.

The same $\hat{Q}_h$ used to obtain the Gibbs policy can also define the Bellman value $\bar{\phi}_h(x)$ implied by one Bellman step in Eq.(5). $\hat{\phi}_h$ is the value predicted by CENO, while $\bar{\phi}_h$ is the value implied by one Bellman step using $\hat{m}_h$ and $\hat{\phi}_{h+1}$. If the predicted value sequence satisfies the backward optimal Bellman relation, then $\hat{\phi}_h(x) = \bar{\phi}_h(x)$. We therefore define the Bellman defect

$$r_h(x) = \hat{\phi}_h(x) - \bar{\phi}_h(x), \quad h = 0, \ldots, H-1. \tag{18}$$

At $h = 0$, $\hat{\phi}_0$ is not a network output because it does not affect any policy; we therefore set $\hat{\phi}_0 := \bar{\phi}_0$, so that $r_0 \equiv 0$. This does not change the policy or population rollout; it only removes a free value component that has no effect on the behavior of the solver.

A direct measure of backward optimality would be a norm of $r_h(x)$, but such a measure depends on value offsets that do not affect the policy: adding a state-independent constant γ_h to each predicted $\hat{\phi}_h$ leaves the Gibbs policy, population rollout, and exploitability unchanged but shifts $r_h(x)$ to $r_h(x) + \gamma_h - \gamma_{h+1}$. The shift relations are proved in Appendix A.2. However, we need a certificate that is invariant to these offsets.

Let $\mathcal{R}_h$ contain the states reachable from $\text{supp}(m_0)$ after h transitions under any admissible policy:

$$\mathcal{R}_0 = \text{supp}(m_0), \quad \mathcal{R}_{h+1} = \{y \in \mathcal{X} : \exists x \in \mathcal{R}_h, a \in \mathcal{A}(x), P_h(y \mid x, a) > 0\}. \tag{19}$$

With finite $\hat{Q}_h$ and $\tau > 0$, the Gibbs policy assigns positive probability to every admissible action. Therefore, $\text{supp}(\hat{m}_h) = \mathcal{R}_h$, as shown in Lemma A.2. For each time step, define

$$C_\theta = \sum_{h=0}^{H-1} C_h, \quad C_h = \max_{x \in \mathcal{R}_h} r_h(x) - \sum_{x \in \mathcal{R}_h} \hat{m}_h(x) r_h(x). \tag{20}$$

Each term compares the largest reachable state defect with the defect averaged under the current population. It is non-negative and unchanged by state-independent shifts of r_h. Further properties are given in Appendix A.3. This form follows from comparing the exact cost $\mathcal{J}_{\hat{m}}(\hat{\pi})$ of the current CENO policy with the best-response cost $\min_\pi \mathcal{J}_{\hat{m}}(\pi)$. For CENO's Gibbs policy, the intermediate value terms cancel, leaving its cost as an initial value term minus its population-weighted defect, as shown in Lemma A.3. The best-response cost is bounded below by the same initial value term minus the sum of the largest reachable state defects, as shown in Lemma A.4. Subtracting these quantities gives the following result:

Theorem 1 (Equilibrium Certificate)**.** *For any network parameters θ and any MFG instance ξ, given the forward-consistent pair $(\hat{\pi}, \hat{m})$ generated by CENO, let $r_h(x)$ denote the Bellman defect defined in Eq.(18) and $R_h(x)$ be the set of reachable states from supp(m_0), and define the certificate C_θ by Eq.(20), then the exploitablity of the generated pair satisfies*

$$0 \le \textit{Expl}_\tau(\hat{\pi}, \hat{m}) = \mathcal{J}_{\hat{m}}(\hat{\pi}) - \min_\pi \mathcal{J}_{\hat{m}}(\pi) \le C_\theta. \tag{21}$$

In particular, $C_\theta = 0$ implies that $(\hat{\pi}, \hat{m})$ is an MFG equilibrium.

The proof is given in Appendix A.4. C_θ can be computed from the predicted values, the induced population, and the known MFG model. It requires neither an equilibrium reference nor the additional backward recursion needed for an exact best response. The converse need not hold: the certificate is an upper bound, not an exact measure of exploitability. Taking a separate maximum at each time step can make this bound conservative.

Appendix A.5 further shows that exploitability equals the population-weighted KL divergence from the exact best response, providing a policy-level interpretation of the certificate.

Training objective. We train CENO by minimizing

$$\mathcal{L}(\theta) = \mathbb{E}_{\xi \sim \mathcal{D}}[C_\theta(\xi)]. \tag{22}$$

By Theorem 1, this objective upper-bounds expected exploitability and requires no equilibrium labels or best-response targets.

4.3 From Individual Optimality to Population-Level Guarantees

Theorem 1 certifies individual optimality by bounding the exploitability of the CENO output, but only at the individual level. This leaves the population trajectory itself unassessed and a near-optimal individual policy does not imply that this trajectory is close to the equilibrium trajectory. We therefore ask whether the same certificate also bounds the distance from $\hat{m}$ to an equilibrium population $m^\star$. The challenge lies in that exploitability compares two policies under one fixed population, whereas the population error compares two populations. The only channel the population enters into an individual's objective is the interaction term $f_h(x, m_h)$, therefore, we start from distinguishing interactions in different population.

For two population distributions $m, m' \in \Delta\mathcal{X}$, define the interaction monotonicity gap

$$\mathcal{M}_h(m, m') = \sum_{x \in \mathcal{X}} [m(x) - m'(x)][f_h(x, m) - f_h(x, m')]. \tag{23}$$

We say that $f_h(x, m_h)$ is monotone if

$$\mathcal{M}_h(m, m') \ge 0, \forall m, m' \in \Delta(\mathcal{X}). \tag{24}$$

Let $(m^\star, \pi^\star)$ be an equilibrium with the same initial population m_0. Comparing the CENO pair $(\hat{m}, \hat{\pi})$ with this equilibrium, and using population-independent transitions, gives

$$\sum_{h=0}^{H-1} \mathcal{M}_h(\hat{m}_h, m_h^\star) + [\mathcal{J}_{m^\star}(\hat{\pi}) - \mathcal{J}_{m^\star}(\pi^\star)] \leq \text{Expl}_\tau(\hat{\pi}, \hat{m}). \tag{25}$$

The detailed derivations are given in Appendix A.7. The first term measures the difference between the two population trajectories through their interaction costs. The second measures the suboptimality of $\hat{\pi}^\star$ under $m^\star$. Under monotonicity, both terms are non-negative, so exploitability also bounds this policy difference. Propagating both state-action pair by the common transition kernel and according Pinsker's inequality, gives:

Theorem 2 (Population-Level Certificate)**.** *Let $(\hat{\pi}, \hat{m})$ be the pair generated by CENO and let C_θ be the equilibrium certificate in Theorem 1. Suppose the population interaction is monotone, i.e.,*

$$\mathcal{M}_h(m, m') \geq 0, \forall m, m' \in \Delta(\mathcal{X}), h = 0, ..., H-1, \tag{26}$$

and let $(\pi^\star, m^\star)$ be an MFG equilibrium with the same initial population m_0. Then

$$\max_{0 \leq h \leq H} \|\hat{m}_h - m_h^\star\|_1 \leq \sqrt{\frac{2\textit{Expl}_\tau(\hat{\pi}, \hat{m})}{\tau}} \leq \sqrt{\frac{2C_\theta}{\tau}}. \tag{27}$$

Thus under monotonicity, the same certificate used to assess individual optimality also controls the deviation of the generated population trajectory from equilibrium. In particular, $C_\theta \to 0$ implies uniform convergence of $\hat{m}_h \to m_h^\star$ over the finite horizon.

Corollary 3 (Accumulated Population Error)**.** *Under the assumptions of Theorem 2, suppose additionally that there exists $\mu > 0$ such that $\mathcal{M}_h(m, m') \geq \mu \|m - m'\|_2^2$ for every h and every $m, m' \in \Delta(\mathcal{X})$. Then*

$$\mu \sum_{h=0}^{H-1} \|\hat{m}_h - m_h^\star\|_2^2 \leq \textit{Expl}_\tau(\hat{\pi}, \hat{m}) \leq C_\theta. \tag{28}$$

Theorem 2 and Corollary 3 give two complementary population-level guarantees: monotonicity bounds the largest trajectory error over time, while strong monotonicity additionally bounds the accumulated squared error across the horizon. Conditions ensuring these properties are discussed in Appendix A.6 and the full derivation is given in Appendix A.7.

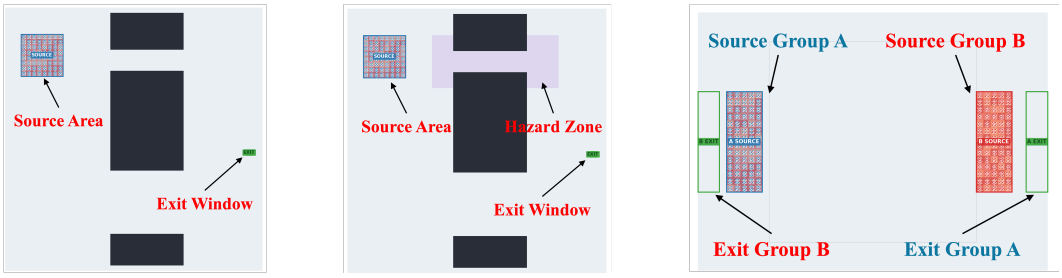
5 Experiment

5.1 Experimental Setup

MFG benchmark. We use a finite-horizon variant of the classical Beach Bar problem in game theory(Magnino et al., 2026). The environment has 16×16 states, 5 actions, and horizon $H = 12$. Transitions are deterministic and population-independent. This benchmark compares CENO with Fixed-Point iteration, Fictitious Play, Online Mirror Descent, and Master Policy in terms of equilibrium quality, online solution time, and certification. Details are given in Appendix B.1.

Multi-agent tasks. We compare CENO with other state-of-the-art multi-agent methods on three multi-agent tasks, shown in Fig. 2, which test three different aspects of population dynamics. All tasks follow the finite-state MFG settings: agents move on a discrete grid with action set $\mathcal{A} = \{\text{up, down, left, right, stay}\}$; each grid cell can accommodate up to 8 agents; the transition depends only on the current state, action, and map layout, and is independent of the population. Thus, all population dependence enters through the interaction term $f_h(x, m_h)$, penalizing local population congestion. Meanwhile, the stage cost has the form $c_h(x, a) + f_h(x, m_h)$, where the term c_h captures task-related preferences such as progress toward the goal, movement cost, and the time-dependent environmental penalties. The horizons in three tasks are unified to $H = 24$. We design three tasks: **(i) spatial allocation:** one population chooses among routes of different widths, balancing a shorter travel distance against higher congestion; **(ii) temporal adaptation:** under a similar spatial layout, a future bottleneck in a hazard zone reduces the capacity of one route, requiring

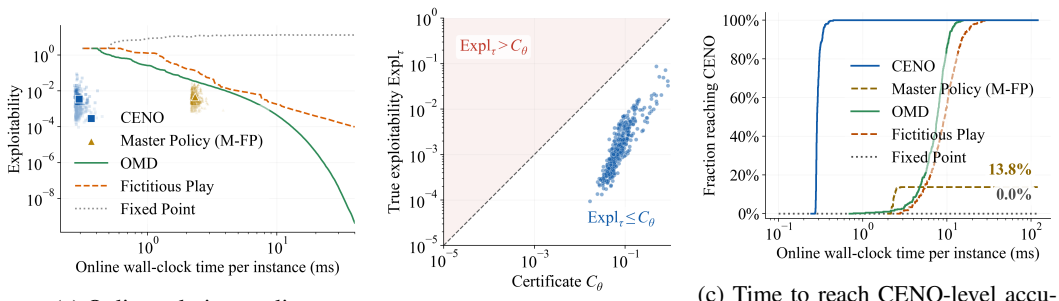
anticipation of future congestion; **(iii) Counterflow coordination:** two populations with opposite goals share the same space and must coordinate their movements. Exact cost definitions, parameter values, metrics, and proofs of strong monotonicity are given in Appendix B.2. We compare CENO with InforMARL(Nayak et al., 2023), MF-PSRO(Muller et al., 2021) and Sable(Mahjoub et al., 2024). These baselines represent graph-based aggregation of local agent information, a mean-field population-based method, and retention-based multi-agent sequence modeling. They are suitable for large-scale tasks. We adapt them to the same grid environments and evaluate them under a common task settings. Implementation details are given in Appendix B.4.



(a) Spatial Allocation (b) Temporal Adaptation (c) Counterflow Coordination

Figure 2: Experimental Setup. (a) Spatial allocation with routes of different widths. (b) Temporal adaptation with a future soft bottleneck that reduces local capacity from 8 to 4 during $h \in [50, 150)$. (c) Counterflow coordination between two populations with opposite goals. The number of agents within each cell is initialized randomly; a new batch of agents is released once all agents have left the source area; blue dots represent individual agents, cell colors indicate occupancy.

5.2 Equilibrium Quality and Certification



(a) Online solution quality (b) Certificate validity (c) Time to reach CENO-level accuracy

Figure 3: **Equilibrium quality and certification on Beach Bar instances (128 instances $\times$ 3 seeds).** (a) Solution quality versus online computation: curves show mean exploitability of iterative methods; small markers represent frozen neural outputs and large markers their means. (b) CENO certificates versus true exploitability. Points in the shaded region would violate $\text{Expl}_\tau \leq C_\theta$. (c) Fraction of instances on which each method reaches the exploitability of the CENO output by each time budget. Online time only includes policy-population construction and CENO certification.

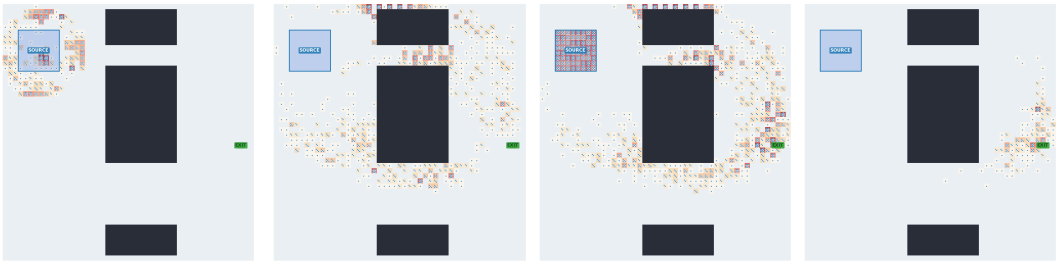
Fig. 3a shows that CENO achieves a mean exploitability of 3.42×10^{-3} with a median online latency of 0.33 ms. Fictitious Play and OMD require approximately $33\times$ and $31\times$ more online time to reach the same per-instance accuracy, although both eventually attain lower exploitability with larger number of iterations. This indicates that CENO's advantage lies in fast near-equilibrium prediction rather than the highest numerical precision. In Fig. 3b, no certificate violation is observed. The Spearman correlation between the certificate and true exploitability is 0.813. Fig. 3c further shows that Fictitious Play and OMD match CENO's accuracy within a 120 ms budget, whereas the Master Policy matches on only 13.8% of instances and Fixed Point on none. Together, CENO solves unseen instances at low cost, with an accompanying error bound.

To examine the performance beyond equilibrium quality results, we provide additional analyses in Appendix B.1. We compare different training objectives, including exact exploitability and the raw Bellman residual, and evaluate whether the certificate-induced population bound is consistent with the population-level error.

5.3 Multi-agent Task Performance

Table 1: Multi-agent performance across spatial allocation, temporal adaptation, and counterflow coordination, respectively. Total Released indicates the realized population load during evaluation. Higher Final Exit Rate and lower Barrier Cost are preferred.

Method	Total Released			Final Exit Rate			Barrier Cost		
Scenario	1	2	3	1	2	3	1	2	3
InforMARL	1745	1825	4204	0.575	0.644	0.437	555.220	286.540	845.431
MF-PSRO	2007	2172	4003	0.864	0.667	0.509	83.821	277.924	77.626
Sable	2013	2320	4180	0.826	0.705	0.331	149.560	342.211	246.974
CENO	2330	2180	3988	0.918	0.780	0.978	25.939	264.527	5.162



(a) $t = 5$ (b) $t = 55$ (c) $t = 150$ (d) $t = 240$

Figure 4: Population evolution under CENO in the spatial allocation. Agents disperse from the source and use multiple routes around the obstacles to reach the exit.

Table 1 shows that CENO achieves the highest final exit rate and lowest barrier cost across all three scenarios. The total released metric measures the actual number of released agents, where a new batch is generated once agents in the source area have completely left. Unlike other methods that move directly toward the exit window, CENO tends to first disperse agents due to local initial density. Consequently, though CENO achieves the largest final exit rate, it does not release the most agents except in the spatial allocation task. The largest advantage appears in the counterflow coordination: CENO achieves a 97.8% exit rate, alongside a negligible barrier cost of 5.162. These results indicate that improved task completion is accompanied by lower congestion.

Fig. 4 illustrates the population evolution in spatial allocation task. Agents initially disperse around the source and subsequently occupy multiple routes around the obstacle, even occupying the one-cell-wide top corridor, which is rarely used by the baselines. These snapshots illustrate distributed route use and progressive evacuation. Additional rollout visualizations for temporal adaptation and counterflow coordination are provided in Appendix B.3.

6 Conclusion

We introduced CENO, a neural solver for finite-state mean-field games. By predicting only the value function, CENO ensures forward consistency by construction and reduces the remaining equilibrium error to a computable certificate, which upper-bounds exploitability without requiring an equilibrium reference or an additional best-response solve. Experiments demonstrate near-equilibrium solutions at low online computational cost and effective population-level coordination. However, CENO currently considers entropy-regularized MFGs with population-independent transition, while the population guarantees further require (strongly) monotone population interaction. Future work will extend the framework to population-dependent dynamics and non-monotone interactions, where both equilibrium characterization and certification become more challenging.

7 AI USAGE STATEMENT

We used generative AI tools for editing the manuscript for readability, and, among tasks with recommended disclosure, for literature search and reference formatting. We did not use them to develop the theoretical framework, formulate or prove the mathematical claims, design the methodology or experiments, or interpret results. We take responsibility for the final content of this work.

8 ETHICS STATEMENT

This work is purely methodological: no human subjects, no personal data, and no released dataset; all instances are synthetic. We note that solvers of this kind target applications such as evacuation and population routing. The certificate bounds the equilibrium gap within the supplied mean-field model and is not a safety guarantee under model misspecification; moreover, a representative-agent formulation does not capture how congestion is distributed across subpopulations. Deployment in human-facing settings requires validation against the real system. The authors adhere to the ICLR Code of Ethics.

9 REPRODUCIBILITY STATEMENT

All assumptions and theoretical derivations are stated in the main text and all claims proved in Appendix A. Experiments require no extra data: Appendix B.1 gives the Beach Bar detailed settings; Appendix B.2 provides the detailed task definitions. Appendix B.4 shows the architecture, training configuration and hardware. We report in the paper the definitions needed to understand and re-derive the method. The full configuration and the complete source code for all experiments and figures are provided as anonymized supplementary material.

REFERENCES

Mouhcine Assouli, Justina Gianatti, Badr Missaoui, and Francisco J Silva. Initialization-driven neural generation and training for high-dimensional optimal control and first-order mean field games. *arXiv preprint arXiv:2507.15126*, 2025.

Cynthia Barnhart, Alexandre Jacquillat, and Alexandria Schmid. Robotic warehousing operations: A learn-then-optimize approach to large-scale neighborhood search. *INFORMS Journal on Optimization*, 7(3):171–194, 2025.

René Carmona and Mathieu Laurière. Convergence analysis of machine learning algorithms for the numerical solution of mean field control and games. *The Annals of Applied Probability*, 32(6): 4065–4105, 2022.

Asaf Cohen, Mathieu Lauriere, and Ethan Zell. Deep backward and galerkin methods for the finite state master equation. *Journal of Machine Learning Research*, 25(401):1–50, 2024.

Kai Cui and Heinz Koeppl. Approximately solving mean field games via entropy-regularized deep reinforcement learning. In *International Conference on Artificial Intelligence and Statistics*, pp. 1909–1917. PMLR, 2021.

Gökçe Dayanıklı and Mathieu Lauriere. Machine learning methods for large population games with applications in operations research. In *Tutorials in Operations Research: Smarter Decisions for a Better World*, pp. 50–89. INFORMS, 2024.

Dena Firoozi, Anastasis Kratsios, and Xuwei Yang. Simultaneously solving infinitely many lq mean field games in hilbert spaces: the power of neural operators. *arXiv preprint arXiv:2510.20017*, 2025.

Jiequn Han, Ruimeng Hu, and Jihao Long. Learning high-dimensional mckean–vlasov forward-backward stochastic differential equations with general distribution dependence. *SIAM Journal on Numerical Analysis*, 62(1):1–24, 2024.

Han Huang and Rongjie Lai. Unsupervised solution operator learning for mean-field games via sampling-invariant parametrizations. *arXiv preprint arXiv:2401.15482*, 2024.

Han Huang and Rongjie Lai. Unsupervised solution operator learning for mean-field games. *Journal of Computational Physics*, 537:114057, 2025.

Kuang Huang, Xu Chen, Xuan Di, and Qiang Du. Dynamic driving and routing games for autonomous vehicles on networks: A mean field game approach. *Transportation Research Part C: Emerging Technologies*, 128:103189, 2021.

Yuhan Kang, Siting Liu, Hongliang Zhang, Wuchen Li, Zhu Han, Stanley Osher, and H Vincent Poor. Joint sensing task assignment and collision-free trajectory optimization for mobile vehicle networks using mean-field games. *IEEE Internet of Things Journal*, 8(10):8488–8503, 2020.

Mathieu Lauriere, Sarah Perrin, Sertan Girgin, Paul Muller, Ayush Jain, Theophile Cabannes, Georgios Piliouras, Julien Pérolat, Romuald Elie, Olivier Pietquin, et al. Scalable deep reinforcement learning algorithms for mean field games. In *International conference on machine learning*, pp. 12078–12095. PMLR, 2022.

Alex Tong Lin, Samy Wu Fung, Wuchen Li, Levon Nurbekyan, and Stanley J Osher. Alternating the population and control neural networks to solve high-dimensional stochastic mean-field games. *Proceedings of the National Academy of Sciences*, 118(31):e2024713118, 2021.

Jialiang Luo and Harry Zheng. Deep neural network solution for finite state mean field game with error estimation. *Dynamic Games and Applications*, 13(3):859–896, 2023.

Lorenzo Magnino, Jiacheng Shen, Matthieu Geist, Olivier Pietquin, and Mathieu Laurière. Bench-mfg: A benchmark suite for learning in stationary mean field games. *arXiv preprint arXiv:2602.12517*, 2026.

Omayma Mahjoub, Sasha Abramowitz, Ruan De Kock, Wiem Khlifi, Simon du Toit, Jemma Daniel, Louay Ben Nessir, Louise Beyers, Claude Formanek, Liam Clark, et al. Sable: a performant, efficient and scalable sequence model for marl. *arXiv preprint arXiv:2410.01706*, 2024.

Paul Muller, Mark Rowland, Romuald Elie, Georgios Piliouras, Julien Perolat, Mathieu Lauriere, Raphael Marinier, Olivier Pietquin, and Karl Tuyls. Learning equilibria in mean-field games: Introducing mean-field psro. *arXiv preprint arXiv:2111.08350*, 2021.

Siddharth Nayak, Kenneth Choi, Wenqi Ding, Sydney Dolan, Karthik Gopalakrishnan, and Hamsa Balakrishnan. Scalable multi-agent reinforcement learning through intelligent information aggregation. In *International conference on machine learning*, pp. 25817–25833. PMLR, 2023.

Julien Perolat, Sarah Perrin, Romuald Elie, Mathieu Laurière, Georgios Piliouras, Matthieu Geist, Karl Tuyls, and Olivier Pietquin. Scaling up mean field games with online mirror descent. *arXiv preprint arXiv:2103.00623*, 2021.

Sarah Perrin, Julien Pérolat, Mathieu Laurière, Matthieu Geist, Romuald Elie, and Olivier Pietquin. Fictitious play for mean field games: Continuous time analysis and applications. *Advances in neural information processing systems*, 33:13199–13213, 2020.

Sarah Perrin, Mathieu Laurière, Julien Pérolat, Romuald Élie, Matthieu Geist, and Olivier Pietquin. Generalization in mean field games by learning master policies. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 36, pp. 9413–9421, 2022.

Lars Ruthotto, Stanley J Osher, Wuchen Li, Levon Nurbekyan, and Samy Wu Fung. A machine learning framework for solving high-dimensional mean field game and mean field control problems. *Proceedings of the National Academy of Sciences*, 117(17):9183–9193, 2020.

Naci Saldi. Linear mean-field games with discounted cost. *Mathematics of Operations Research*, 51(1):358–397, 2026.

Qing Tang and Jiahao Song. Learning optimal policies in potential mean field games: Smoothed policy iteration algorithms. *SIAM Journal on Control and Optimization*, 62(1):351–375, 2024.

Anna CM Thöni, Yoram Bachrach, and Tal Kachman. Neural mfgs: Extending mean-field game theory with neural stochastic differential equations. In *Proc. of the 25th International Conference on Autonomous Agents and Multiagent Systems*, pp. 3368–3370, 2026.

Noureddine Toumi, Roland Malhame, and Jerome Le Ny. A mean field game approach for a class of linear quadratic discrete choice problems with congestion avoidance. *Automatica*, 160:111420, 2024.

Yutong Wang, Bairan Xiang, Shinan Huang, and Guillaume Sartoretti. Scrimp: Scalable communication for reinforcement-and imitation-learning-based multi-agent pathfinding. In *2023 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pp. 9301–9308. IEEE, 2023.

Runtian Zeng, Chuandong Li, Lixin Li, Mengqi Li, and Tingkai Hu. Solving mean field game based on physics-informed operator learning and deep reinforcement learning. *Journal of Computational Physics*, pp. 114457, 2025.

A Theoretical Appendix

A.1 Exact Forward Consistency

We provide the proof of exact forward consistency here. First, we give several basic properties of the policy-induced Markov dynamics.

Lemma A.1 (Properties of the policy-induced Markov dynamics)**.** *For any stochastic policy π_h, let M_{π_h} be its induced transition matrix. Then M_{π_h} is non-negative and row-stochastic, and the population update $m_{h+1} = m_h M_{\pi_h}$ satisfies the following properties:*

1. *Non-negative: $m_h \geq 0 \Rightarrow m_{h+1} \geq 0$;*
2. *Mass conservation: $\sum_y m_{h+1}(y) = \sum_x m_h(x)$, therefore $m_h \in \Delta(\mathcal{X}) \Rightarrow m_{h+1} \in \Delta(\mathcal{X})$;*
3. *ℓ^1 non-expansion: $\|zM_{\pi_h}\|_1 \leq \|z\|_1,\ \forall z \in \mathbb{R}^S$.*

Proof. Since both $\pi_h(a \mid x)$ and $P_h(y \mid x, a)$ are non-negative,

$$M_{\pi_h}(x, y) = \sum_{a \in \mathcal{A}(x)} \pi_h(a \mid x) P_h(y \mid x, a) \geq 0. \tag{29}$$

For every x,

$$\begin{aligned} \sum_{y \in \mathcal{X}} M_{\pi_h}(x, y) &= \sum_y \sum_a \pi_h(a \mid x) P_h(y \mid x, a) \\ &= \sum_a \pi_h(a \mid x) \sum_y P_h(y \mid x, a) \\ &= \sum_a \pi_h(a \mid x) \\ &= 1, \end{aligned} \tag{30}$$

so M_{π_h} is row-stochastic.

Non-negativity of $m_h M_{\pi_h}$ follows directly whenever $m_h \geq 0$. For mass conservation,

$$\begin{aligned} \sum_y (m_h M_{\pi_h})(y) &= \sum_y \sum_x m_h(x) M_{\pi_h}(x, y) \\ &= \sum_x m_h(x) \sum_y M_{\pi_h}(x, y) \\ &= \sum_x m_h(x). \end{aligned} \tag{31}$$

Hence a probability distribution remains in $\Delta(\mathcal{X})$ after the update.

Finally, for any $z \in \mathbb{R}^S$,

$$\begin{aligned}\|zM_{\pi_h}\|_1 &= \sum_y \left| \sum_x z(x) M_{\pi_h}(x,y) \right| \\ &\le \sum_y \sum_x |z(x)| M_{\pi_h}(x,y) \\ &= \sum_x |z(x)| \sum_y M_{\pi_h}(x,y) \\ &= \sum_x |z(x)| \\ &= \|z\|_1. \end{aligned} \tag{32}$$

□

Then, we give the proof of forward consistency.

Proof. For every state x, the Gibbs policy generated by CENO satisfies

$$\hat{\pi}_h(a \mid x) \ge 0, \quad \sum_{a \in \mathcal{A}(x)} \hat{\pi}_h(a \mid x) = 1. \tag{33}$$

Suppose $\hat{m}_h \in \Delta(\mathcal{X})$. Since $M_{\hat{\pi}_h}$ is row-stochastic, Lemma A.1 gives,

$$\hat{m}_{h+1} = \hat{m}_h M_{\hat{\pi}_h} \in \Delta(\mathcal{X}). \tag{34}$$

Starting from $\hat{m}_0 = m_0$, induction therefore gives

$$\hat{m}_h \in \Delta(\mathcal{X}), \quad h = 0, \dots, H. \tag{35}$$

Moreover, the relation

$$\hat{m}_{h+1} = \hat{m}_h M_{\hat{\pi}_h} \tag{36}$$

is the definition of the population rollout in CENO. Therefore the forward residual satisfies

$$R_h^{\text{fwd}} = \hat{m}_{h+1} - \hat{m}_h M_{\hat{\pi}_h} = 0, \quad h = 0, \dots, H-1. \tag{37}$$

The equality holds for every MFG instance ξ and every network parameter θ, independently of the accuracy of the predicted continuation values. □

Lemma A.2 (Reachable set equals population support)**.** *Define*

$$\mathcal{R}_0 = \mathit{supp}(m_0) \tag{38}$$

and recursively

$$\mathcal{R}_{h+1} = \{y \in \mathcal{X} : \exists x \in \mathcal{R}_h, a \in \mathcal{A}(x) \textit{ such that } P_h(y \mid x, a) > 0\}. \tag{39}$$

For the population trajectory generated by CENO,

$$\mathit{supp}(\hat{m}_h) = \mathcal{R}_h, \quad h = 0, \dots, H. \tag{40}$$

Proof. We proceed by induction. At $h = 0$,

$$\text{supp}(\hat{m}_0) = \text{supp}(m_0) = \mathcal{R}_0. \tag{41}$$

Suppose that

$$\text{supp}(\hat{m}_h) = \mathcal{R}_h. \tag{42}$$

The next population satisfies

$$\hat{m}_{h+1}(y) = \sum_{x \in \mathcal{X}} \sum_{a \in \mathcal{A}(x)} \hat{m}_h(x) \hat{\pi}_h(a \mid x) P_h(y \mid x, a). \tag{43}$$

Every term in this sum is non-negative. Since $\tau > 0$ and the Q-values are finite, the Gibbs policy assigns strictly positive probability to every admissible action:

$$\hat{\pi}_h(a \mid x) > 0, \quad \forall a \in \mathcal{A}(x). \tag{44}$$

Therefore, $\hat{m}_{h+1}(y) > 0$ if and only if there exist x and a such that

$$\hat{m}_h(x) > 0 \quad \text{and} \quad P_h(y \mid x, a) > 0. \tag{45}$$

By the induction hypothesis,

$$\hat{m}_h(x) > 0 \Leftrightarrow x \in \mathcal{R}_h. \tag{46}$$

Hence,

$$\hat{m}_{h+1}(y) > 0 \Leftrightarrow y \in \mathcal{R}_{h+1}. \tag{47}$$

Thus,

$$\text{supp}(\hat{m}_{h+1}) = \mathcal{R}_{h+1}, \tag{48}$$

which completes the induction. □

The reachable sets are defined by the initial support and the transition model, not by a numerical threshold on the rollout population. A state with a small positive probability remains part of the reachable set used by the certificate.

A.2 Value Shifts and Policy Invariance

Fix a population distribution m_h. Consider a continuation-value shift

$$\phi'_{h+1}(y) = \phi(y)_{h+1} + \gamma_{h+1}, \quad \forall y \in \mathcal{X}, \tag{49}$$

where γ_{h+1} is independent of the state. The corresponding state-action value is

$$Q'_h(x, a) = c_h(x, a) + f_h(x, m_h) + \sum_y P_h(y \mid x, a)\phi'_{h+1}(y). \tag{50}$$

Since $\sum_y P_h(y \mid x, a) = 1$, we obtain

$$Q'_h(x, a) = Q_h(x, a) + \gamma_{h+1}. \tag{51}$$

The corresponding Bellman value satisfies

$$\begin{aligned} \bar{\phi}'_h(x) &= -\tau \log \sum_a \exp\left(-\frac{Q_h(x, a) + \gamma_{h+1}}{\tau}\right) \\ &= \gamma_{h+1} - \tau \log \sum_a \exp\left(-\frac{Q_h(x, a)}{\tau}\right) \\ &= \bar{\phi}_h(x) + \gamma_{h+1}. \end{aligned} \tag{52}$$

For the Gibbs policy,

$$\begin{aligned} \pi'_h(a \mid x) &= \frac{\exp\left(-\frac{Q_h(x,a)+\gamma_{h+1}}{\tau}\right)}{\sum_b \exp\left(-\frac{Q_h(x,b)+\gamma_{h+1}}{\tau}\right)} \\ &= \frac{\exp\left(-\frac{Q_h(x,a)}{\tau}\right)}{\sum_b \exp\left(-\frac{Q_h(x,b)}{\tau}\right)} \\ &= \pi_h(a \mid x). \end{aligned} \tag{53}$$

Therefore, adding a state-independent constant to the continuation value changes neither the Gibbs policy nor the population generated by that policy.

Now consider the predicted CENO sequence

$$\hat{\phi}'_h(x) = \hat{\phi}_h(x) + \gamma_h, \quad h = 1, \ldots, H-1, \tag{54}$$

with $\gamma_H = 0$ because the terminal value is fixed by $\hat{\phi}_H = g$. At time h,

$$\hat{Q}'_h(x,a) = \hat{Q}_h(x,a) + \gamma_{h+1} \tag{55}$$

According to Eq.(5), the Bellman values satisfy

$$\bar{\phi}'_h(x) = \bar{\phi}_h(x) + \gamma_{h+1}. \tag{56}$$

For $h = 1, \ldots, H-1$,

$$\begin{aligned} r'_h(x) &= \hat{\phi}'_h(x) - \bar{\phi}'_h(x) \\ &= \hat{\phi}_h(x) + \gamma_h - \bar{\phi}_h(x) - \gamma_{h+1} \\ &= r_h(x) + \gamma_h - \gamma_{h+1}. \end{aligned} \tag{57}$$

At $h = 0$, CENO always sets $\hat{\phi}_0 := \bar{\phi}_0$ and therefore $r_0 = 0$. After shifting $\hat{\phi}_1$, both $\bar{\phi}_0$ and the resulting $\hat{\phi}_0$ shift by the same amount, so $r'_0 = 0$. Therefore, state-independent offsets in the predicted values can change the Bellman defects without changing the policy, population trajectory, or exploitability.

More generally, adding the same quantity to every action value at a given state leaves the Gibbs policy unchanged. For any function $b_h(x)$,

$$\frac{\exp[-(Q_h(x,a) + b_h(x))/\tau]}{\sum_{a' \in \mathcal{A}(x)} \exp[-(Q_h(x,a') + b_h(x))/\tau]} = \frac{\exp[-Q_h(x,a)/\tau]}{\sum_{a' \in \mathcal{A}(x)} \exp[-Q_h(x,a')/\tau]} \tag{58}$$

In particular, the action-independent interaction $f_h(x, \hat{m}_h)$ cancels from the normalization used to construct $\hat{\pi}_h$. It remains part of Bellman value and therefore contributes to the Bellman defect.

A.3 Basic Properties of the Certificate

We use the largest Bellman defect over the reachable states and define the certificate at h as

$$C_h = \max_{x \in \mathcal{R}_h} r_h(x) - \sum_{x \in \mathcal{R}_h} \hat{m}_h(x) r_h(x). \tag{59}$$

By Lemma A.2, $\mathrm{supp}(\hat{m}_h) = \mathcal{R}_h$. Thus, $\sum_{x \in \mathcal{R}_h} \hat{m}_h(x) = 1$, and every state in $\mathcal{R}_h$ has positive population mass.

Proposition 4 (Basic properties of the certificate)**.** *For every time step h, the certificate C_h has the following properties:*

1. *Non-negative: $C_h \geq 0$;*
2. *Bounded by the defect range: $C_h \leq \max_{x \in \mathcal{R}_h} r_h(x) - \min_{x \in \mathcal{R}_h} r_h(x)$;*
3. *Shift invariant: $C'_h = C_h$;*
4. *Zero certificate: $C_h = 0$ iff r_h is constant over $\mathcal{R}_h$.*

Proof. Because $\hat{m}_h$ is a probability distribution supported on $\mathcal{R}_h$,

$$\min_{x \in \mathcal{R}_h} r_h(x) \leq \sum_{x \in \mathcal{R}_h} \hat{m}_h(x) r_h(x) \leq \max_{x \in \mathcal{R}_h} r_h(x). \tag{60}$$

The first two claims follow directly.

Now let $r'_h(x) = r_h(x) + \beta_h$ for all $x \in \mathcal{R}_h$. Then

$$\max_{x \in \mathcal{R}_h} r'_h(x) = \max_{x \in \mathcal{R}_h} r_h(x) + \beta_h, \tag{61}$$

and

$$\begin{aligned} \sum_{x \in \mathcal{R}_h} \hat{m}_h(x) r'_h(x) &= \sum_{x \in \mathcal{R}_h} \hat{m}_h(x)[r_h(x) + \beta_h] \\ &= \sum_{x \in \mathcal{R}_h} \hat{m}_h(x) r_h(x) + \beta_h \end{aligned} \tag{62}$$

Substituting these expressions into the definition of C_h gives

$$C'_h = C_h. \tag{63}$$

Since every reachable state has positive mass, the weighted average equals the maximum if and only if all reachable states have the same Bellman defect. Thus,

$$C_h = 0 \quad \Leftrightarrow \quad r_h(x) \text{ is constant on } \mathcal{R}_h. \tag{64}$$

The maximum is taken separately at each time step. This bounds the defect encountered by every admissible policy, but a single policy need not reach a maximizing state at every time. The sum can therefore overestimate the actual gain from deviation. In particular, a positive certificate does not by itself imply that the policy has a large exploitability. □

A.4 Proof of Equilibrium Certificate

By definition,

$$\mathrm{Expl}_\tau(\hat{\pi}, \hat{m}) = \mathcal{J}_{\hat{m}}(\hat{\pi}) - \min_{\pi} \mathcal{J}_{\hat{m}}(\pi), \tag{65}$$

we first derive an exact expression from the current policy cost $\mathcal{J}_{\hat{m}}(\hat{\pi})$ and then a lower bound on the best-response cost $\min_{\pi} \mathcal{J}_{\hat{m}}(\pi)$.

Exact cost identity. The Gibbs policy and the forward rollout give the following identity.

Lemma A.3 (Exact cost identity)**.**

$$\mathcal{J}_{\hat{m}}(\hat{\pi}) = \sum_x m_0(x)\hat{\phi}_0(x) - \sum_{h=0}^{H-1} \sum_{x \in \mathcal{R}_h} \hat{m}_h(x) r_h(x). \tag{66}$$

Proof. We first give some relations that will be used below. For any finite vector $q = (q_1, \ldots, q_K)$, let

$$p_i = \frac{e^{-q_i/\tau}}{\sum_j e^{-q_j/\tau}}. \tag{67}$$

Then

$$\log p_i = -\frac{q_i}{\tau} - \log \sum_j e^{-q_j/\tau}, \tag{68}$$

which gives

$$\sum_i p_i (q_i + \tau \log p_i) = -\tau \log \sum_i e^{-q_i/\tau}. \tag{69}$$

Applying this identity to $\hat{Q}_h(x, \cdot)$ and the Gibbs policy $\hat{\pi}_h(\cdot \mid x)$, and using the definition of $\bar{\phi}_h$ and r_h, gives

$$\begin{aligned} \sum_{a \in \mathcal{A}(x)} \hat{\pi}_h(a \mid x) \left[\hat{Q}_h(x,a) + \tau \log \hat{\pi}_h(a \mid x) \right] &= -\tau \log \sum_{a \in \mathcal{A}(x)} \exp\left(-\frac{\hat{Q}_h(x,a)}{\tau} \right) \\ &= \bar{\phi}_h(x) \\ &= \hat{\phi}_h(x) - r_h(x). \end{aligned} \tag{70}$$

Multiply both sides by $\hat{m}_h(x)$ and sum over x. Expanding $\hat{Q}_h$ gives

$$\begin{aligned} &\sum_{x,a} \hat{m}_h(x) \hat{\pi}_h(a \mid x) \left[\hat{Q}_h(x,a) + \tau \log \hat{\pi}_h(a \mid x) \right] \\ =& \sum_{x,a} \hat{m}_h(x) \hat{\pi}_h(a \mid x) \left[c_h(x,a) + f_h(x, \hat{m}_h) + \tau \log \hat{\pi}_h(a \mid x) \right] \\ &+ \sum_{x,a,y} \hat{m}_h(x) \hat{\pi}_h(a \mid x) P_h(y \mid x, a) \hat{\phi}_{h+1}(y) \\ =& \sum_x \hat{m}_h(x) \left[\hat{\phi}_h(x) - r_h(x) \right]. \end{aligned} \tag{71}$$

By exact forward consistency,

$$\hat{m}_{h+1}(y) = \sum_{x,a} \hat{m}_h(x)\hat{\pi}_h(a \mid x)P_h(y \mid x, a), \tag{72}$$

so

$$\sum_{x,a,y} \hat{m}_h(x)\hat{\pi}_h(a \mid x)P_h(y \mid x, a)\hat{\phi}_{h+1}(y) = \sum_y \hat{m}_{h+1}(y)\hat{\phi}_{h+1}(y). \tag{73}$$

Let Λ_h denote the expected entropy-regularized stage cost at time h. We obtain

$$\begin{aligned}\Lambda_h &= \sum_{x,a} \hat{m}_h(x)\hat{\pi}_h(a \mid x)\left[c_h(x, a) + f_h(x, \hat{m}_h) + \tau \log \hat{\pi}_h(a \mid x)\right] \\ &= \sum_x \hat{m}_h(x)\hat{\phi}_h(x) - \sum_x \hat{m}_h(x)r_h(x) - \sum_y \hat{m}_{h+1}(y)\hat{\phi}_{h+1}(y).\end{aligned} \tag{74}$$

Summing over $h = 0, \ldots, H-1$, all intermediate value terms cancel. Hence,

$$\sum_{h=0}^{H-1} \Lambda_h = \sum_x m_0(x)\hat{\phi}_0(x) - \sum_{h=0}^{H-1}\sum_x \hat{m}_h(x)r_h(x) - \sum_x \hat{m}_H(x)\hat{\phi}_H(x). \tag{75}$$

Since $\hat{\phi}_H = g$, adding the terminal cost $\sum_x \hat{m}_H(x)g(x)$ cancels the last term. Therefore, the objective in Eq.(3) becomes

$$\mathcal{J}_{\hat{m}}(\hat{\pi}) = \sum_x m_0(x)\hat{\phi}_0(x) - \sum_{h=0}^{H-1}\sum_{x\in\mathcal{R}_h} \hat{m}_h(x)r_h(x). \tag{76}$$

□

Lower bound on the best-response cost. Then, for $\min_\pi \mathcal{J}_{\hat{m}}(\pi)$, we can derive

Lemma A.4 (Lower bound on the exact best-response cost)**.**

$$\min_\pi \mathcal{J}_{\hat{m}}(\pi) \geq \sum_x m_0(x)\hat{\phi}_0(x) - \sum_{h=0}^{H-1} \max_{x\in\mathcal{R}_h} r_h(x). \tag{77}$$

Proof. Define $s_H = 0$, and for $h = 0, \ldots, H-1$, $s_h = \sum_{t=h}^{H-1} \max_{x\in\mathcal{R}_t} r_t(x)$.

We construct the shifted value sequence

$$\tilde{\phi}_h(x) = \hat{\phi}_h(x) - s_h. \tag{78}$$

For every $x \in \mathcal{R}_h$,

$$r_h(x) = \hat{\phi}_h(x) - \bar{\phi}_h(x) \leq \max_{x\in\mathcal{R}_h} r_h(x) \tag{79}$$

Hence

$$\hat{\phi}_h(x) - \max_{x\in\mathcal{R}_h} r_h(x) \leq \bar{\phi}_h(x) \tag{80}$$

Since $s_h = \max_{x\in\mathcal{R}_h} r_h(x) + s_{h+1}$, we obtain

$$\tilde{\phi}_h(x) \leq \bar{\phi}_h(x) - s_{h+1} \tag{81}$$

Now replace the continuation value $\hat{\phi}_{h+1}$ by

$$\tilde{\phi}_{h+1}(y) = \hat{\phi}_{h+1}(y) - s_{h+1}. \tag{82}$$

From Appendix A.2, this shifts every corresponding Q-value and its Bellman value by $-s_{h+1}$. Therefore, $\bar{\phi}_h(x) - s_{h+1}$ is exactly the Bellman value obtained from the continuation value $\tilde{\phi}_{h+1}$.

Let $\phi_h^{\hat{m}}$ denote the exact best-response value from the fixed population trajectory $\hat{m}$. It satisfies soft Bellman recursion, and $\phi_H^{\hat{m}} = g$. At the terminal step,

$$\tilde{\phi}_H = \hat{\phi}_H = g = \phi_H^{\hat{m}}. \tag{83}$$

Assume that $\tilde{\phi}_{h+1}(y) \le \phi_{h+1}^{\hat{m}}(y)$ for every $y \in \mathcal{R}_{h+1}$. For any $x \in \mathcal{R}_h$, all successor states with positive transition probability belong to $\mathcal{R}_{h+1}$. Hence, for every admissible action,

$$\sum_y P_h(y \mid x, a)\tilde{\phi}_{h+1}(y) \le \sum_y P_h(y \mid x, a)\phi_{h+1}^{\hat{m}}(y). \tag{84}$$

The corresponding state-action values therefore satisfy the same ordering. Since the soft Bellman value is non-decreasing in each state-action value, the Bellman value obtained from $\tilde{\phi}_{h+1}$ is no larger than $\phi_h^{\hat{m}}(x)$.

Combining this with the previous inequality gives

$$\tilde{\phi}_h(x) \le \phi_h^{\hat{m}}(x), \quad x \in \mathcal{R}_h. \tag{85}$$

Backward induction therefore yields

$$\tilde{\phi}_h(x) \le \phi_h^{\hat{m}}(x), \quad x \in \mathcal{R}_h, \quad h = 0, \dots, H. \tag{86}$$

At $h = 0$, $\mathcal{R}_0 = \text{supp}(m_0)$, and hence

$$\phi_0^{\hat{m}}(x) \ge \hat{\phi}_0(x) - \sum_{h=0}^{H-1} \max_{x \in \mathcal{R}_h} r_h(x) \tag{87}$$

for every state with positive initial population mass. Therefore,

$$\min_\pi \mathcal{J}_{\hat{m}}(\pi) = \sum_x m_0(x)\phi_0^{\hat{m}}(x) \ge \sum_x m_0(x)\hat{\phi}_0(x) - \sum_{h=0}^{H-1} \max_{x \in \mathcal{R}_h} r_h(x). \tag{88}$$

□

Proof of Equilibrium Certificate. Theorem 1 follows directly from Lemmas A.3 and A.4:

Proof. From Lemma. A.3,

$$\mathcal{J}_{\hat{m}}(\hat{\pi}) = \sum_x m_0(x)\hat{\phi}_0(x) - \sum_{h=0}^{H-1} \sum_{x \in \mathcal{R}_h} \hat{m}_h(x) r_h(x). \tag{89}$$

From Lemma. A.4,

$$\min_\pi \mathcal{J}_{\hat{m}}(\pi) \ge \sum_x m_0(x)\hat{\phi}_0(x) - \sum_{h=0}^{H-1} \max_{x \in \mathcal{R}_h} r_h(x). \tag{90}$$

Therefore,

$$\begin{aligned} \text{Expl}_\tau(\hat{\pi}, \hat{m}) &= \mathcal{J}_{\hat{m}}(\hat{\pi}) - \min_\pi \mathcal{J}_{\hat{m}}(\pi) \\ &\le \sum_{h=0}^{H-1} \left[\max_{x \in \mathcal{R}_h} r_h(x) - \sum_{x \in \mathcal{R}_h} \hat{m}_h(x) r_h(x) \right] \\ &= C_\theta \end{aligned} \tag{91}$$

The lower bound $\text{Expl}_\tau(\hat{\pi}, \hat{m}) \ge 0$ follows directly from the definition of exploitability. Finally,

$$0 \le \text{Expl}_\tau(\hat{\pi}, \hat{m}) \le C_\theta. \tag{92}$$

This proves Theorem 1. □

This proof does not require monotonicity of the population interaction. The ordering used in the backward comparison follows from the soft Bellman value being increasing in each action value.

A.5 POLICY-LEVEL INTERPRETATION

Theorem 1 bounds the exploitability of the CENO output. We first state a policy cost identity for a fixed population trajectory and then apply it to CENO.

Fix a population trajectory m, and let $\pi^{\mathrm{BR},m}$ be its exact entropy-regularized best response. For an arbitrary policy π, let d_h^π denote the state distribution generated by π from m_0.

Proposition 5 (Policy Cost Gap and KL Divergence)**.**

$$\mathcal{J}_m(\pi) - \min_{\pi'} \mathcal{J}_m(\pi') = \tau \sum_{h=0}^{H-1} \sum_x d_h^\pi(x) KL\left(\pi_h(\cdot \mid x) \| \pi_h^{BR,m}(\cdot \mid x)\right). \tag{93}$$

This identity does not require $d^\pi = m$. When the pair (π, m) is forward-consistent, the cost gap on the left is its exploitability.

Proof. Let ϕ_h^m denote the exact best-response value for the fixed population trajectory m, and define

$$Q_h^{\star,m}(x,a) = c_h(x,a) + f_h(x, m_h) + \sum_y P_h(y \mid x, a)\phi_{h+1}^m(y). \tag{94}$$

The exact best response is

$$\pi_h^{\mathrm{BR},m}(a \mid x) = \frac{\exp(-Q_h^{\star,m}(x,a)/\tau)}{\sum_b \exp(-Q_h^{\star,m}(x,b)/\tau)}. \tag{95}$$

For any policy distribution $\pi_h(\cdot \mid x)$,

$$\begin{aligned} &\sum_a \pi_h(a \mid x)[Q_h^{\star,m}(x,a) + \tau \log \pi_h(a \mid x)] - \phi_h^m(x) \\ &\qquad = \tau \mathrm{KL}(\pi_h(\cdot \mid x) \| \pi_h^{\mathrm{BR},m}(\cdot \mid x)). \end{aligned} \tag{96}$$

After multiplying by $d_h^\pi(x)$ and summing over states and time, the continuation-value terms telescope because d_h^π is the state flow generated by π. The left-hand side becomes

$$\mathcal{J}_m(\pi) - \sum_x m_0(x)\phi_0^m(x). \tag{97}$$

Since

$$\sum_x m_0(x)\phi_0^m(x) = \min_{\pi'} \mathcal{J}_m(\pi'), \tag{98}$$

we obtain

$$\mathcal{J}_m(\pi) - \min_{\pi'} \mathcal{J}_m(\pi') = \tau \sum_{h,x} d_h^\pi(x) \mathrm{KL}\left(\pi_h(\cdot \mid x) \| \pi_h^{\mathrm{BR},m}(\cdot \mid x)\right). \tag{99}$$

□

This identity shows that entropy-regularized exploitability is exactly the accumulated policy difference from the best response, weighted by the state actually visited by the current policy. For CENO, exact forward consistency gives $d_h^{\hat{\pi}} = \hat{m}_h$. Therefore, combining this identity with Theorem 1, Eq.(93) becomes

$$\mathrm{Expl}_\tau(\hat{\pi}, \hat{m}) = \tau \sum_{h,x} \hat{m}_h(x) \mathrm{KL}\left(\hat{\pi}_h(\cdot \mid x) \| \pi_h^{\mathrm{BR},\hat{m}}(\cdot \mid x)\right) \le C_\theta. \tag{100}$$

The KL bound can further be converted into a direct distance between action distributions. By Pinsker's inequality,

$$\mathrm{KL}\left(\hat{\pi}_h(\cdot \mid x) \| \pi_h^{\mathrm{BR},\hat{m}}(\cdot \mid x)\right) \ge \frac{1}{2} \|\hat{\pi}_h(\cdot \mid x) - \pi_h^{\mathrm{BR},\hat{m}}(\cdot \mid x)\|_1^2. \tag{101}$$

Together, these results give a policy-level meaning to the certificate:

$$\frac{\tau}{2} \sum_{h,x} \hat{m}_h(x) \|\hat{\pi}_h(\cdot \mid x) - \pi_h^{\mathrm{BR},\hat{m}}(\cdot \mid x)\|_1^2 \le \mathrm{Expl}_\tau(\hat{\pi}, \hat{m}) \le C_\theta. \tag{102}$$

A.6 Monotonicity and Strong Monotonicity

For two population distributions $m, m' \in \Delta(\mathcal{X})$, define the interaction difference

$$\mathcal{M}_h(m, m') = \sum_{x \in \mathcal{X}} [m(x) - m'(x)][f_h(x, m) - f_h(x, m')]. \tag{103}$$

Ordinary monotonicity. The Lasry-Lions monotonicity condition is

$$\mathcal{M}_h(m, m') \geq 0. \tag{104}$$

The condition does not require the interaction difference to be strictly positive when $m \neq m'$. For example, if $f_h(x, m) = b_h(x)$ does not depend on the population, then $\mathcal{M}_h(m, m') = 0$ for every pair of distributions. Thus, ordinary monotonicity alone does not give a positive quadratic lower bound on $\mathcal{M}_h$.

Strict monotonicity. Strict monotonicity strengthens this condition to

$$\mathcal{M}_h(m, m') > 0, \quad m \neq m'. \tag{105}$$

It distinguishes different populations through their interaction costs. Consider

$$m = (p, 1-p), \quad m' = (q, 1-q), \tag{106}$$

with

$$f_h(1, m) = p^3, \quad f_h(2, m) = 0. \tag{107}$$

For $p \neq q$,

$$\mathcal{M}_h(m, m') = (p-q)(p^3 - q^3) > 0, \tag{108}$$

while

$$\frac{\mathcal{M}_h(m, m')}{\|m - m'\|_2^2} = \frac{p^2 + pq + q^2}{2}. \tag{109}$$

Taking $q = 0$ and $p \to 0$, this relation approaches zero. Hence no uniform $\mu > 0$ satisfies

$$\mathcal{M}_h(m, m') \geq \mu \|m - m'\|_2^2 \tag{110}$$

for all populations in this example. Strict monotonicity therefore gives separation between different populations, but not the uniform lower bound required for a metric error estimate.

Strong monotonicity. Strong monotonicity requires a constant $\mu > 0$, independent of h, m, m', such that

$$\mathcal{M}_h(m, m') \geq \mu \|m - m'\|_2^2, \quad \forall m, m' \in \Delta(\mathcal{X}). \tag{111}$$

It directly relates the interaction difference to the squared population distance.

The following examples give sufficient conditions for strong monotonicity.

Local linear congestion. Consider

$$f_h(x, m) = \kappa_h(x) m(x), \quad \text{where } \kappa_h(x) \geq \underline{\kappa} > 0. \tag{112}$$

In this case,

$$\mathcal{M}_h(m, m') = \sum_x \kappa_h(x)[m(x) - m'(x)]^2 \geq \underline{\kappa} \|m - m'\|_2^2, \tag{113}$$

so the interaction is strongly monotone with $\mu = \underline{\kappa}$.

Nonlinear separable congestion. Suppose

$$f_h(x, m) = g_{h,x}(m(x)), \tag{114}$$

where each $g_{h,x}$ is differentiable and satisfies

$$g'_{h,x}(s) \geq \mu > 0, \quad s \in [0, 1]. \tag{115}$$

By the mean value theorem,

$$[m(x) - m'(x)][g_{h,x}(m(x)) - g_{h,x}(m'(x))] \geq \mu[m(x) - m'(x)]^2. \tag{116}$$

Summing over states gives

$$\mathcal{M}_h(m, m') \geq \mu\|m - m'\|_2^2. \tag{117}$$

Hence any uniformly increasing separable congestion cost satisfies the required condition.

Differentiability is sufficient but not necessary for this conclusion. For example, suppose

$$f_h(x, m) = \kappa_h(x)m(x) + b_{h,x}(m(x)), \quad \kappa_h \geq \underline{\kappa} > 0, \tag{118}$$

where $b_{h,x}$ is non-decreasing. Then

$$\begin{aligned}\mathcal{M}_h(m, m') =& \sum_x \kappa_h(x)[m(x) - m'(x)]^2 \\ &+ \sum_x [m(x) - m'(x)][b_{h,x}(m(x)) - b_{h,x}(m'(x))] \\ \geq& \underline{\kappa}\|m - m'\|_2^2.\end{aligned} \tag{119}$$

This argument also applies to non-decreasing congestion functions with corners or caps.

General linear interaction. A more general case is a linear interaction of the form

$$f_h(m) = A_h m, \tag{120}$$

where $f_h(m) \in \mathbb{R}^S$ collects the interaction costs over all states. Let $z = m - m'$. Since both m and m' are probability distributions, $\sum_x z(x) = 0$.

The interaction difference becomes

$$\mathcal{M}_h(m, m') = z^\top A_h z. \tag{121}$$

Only the symmetric part of A_h contributes to this quadratic form:

$$z^\top A_h z = z^\top \frac{A_h + A_h^\top}{2} z. \tag{122}$$

Therefore, a sufficient condition for strong monotonicity is

$$z^\top \frac{A_h + A_h^\top}{2} z \geq \mu\|z\|_2^2 \tag{123}$$

for every population-difference direction z satisfying $\sum_x z(x) = 0$. In particular, A_h itself does not need to be symmetric; only its symmetric part needs to have a positive lower bound on the zero-sum subspace $\{z : \sum_x z(x) = 0\}$.

More generally, suppose that f_h is continuously differentiable on a neighborhood of the simplex. A sufficient condition for strong monotonicity is a uniform positive lower bound on the symmetric part of its Jacobian along population-difference directions. Let

$$J_h(m) = \frac{\partial f_h(m)}{\partial m}. \tag{124}$$

If, for every $m \in \Delta(\mathcal{X})$ and every z satisfying $\sum_x z(x) = 0$,

$$z^\top \frac{J_h(m) + J_h(m)^\top}{2} z \geq \mu\|z\|_2^2, \tag{125}$$

then, using

$$f_h(m) - f_h(m') = \int_0^1 J_h(m' + t(m - m'))(m - m')dt, \tag{126}$$

we obtain

$$\mathcal{M}_h(m, m') \geq \mu\|m - m'\|_2^2. \tag{127}$$

This Jacobian condition provides a convenient way to check strong monotonicity for more general interaction models.

A.7 Population-Level Guarantee

Throughout this section, we consider the same finite-horizon instance with finite costs, entropy coefficient $\tau > 0$, population-independent transition kernel P_h, and terminal cost g. Let $(\hat{\pi}, \hat{m})$ be a forward-consistent pair, and let $(\pi^\star, m^\star)$ be an equilibrium with the same initial distribution m_0. We take $\pi^\star$ to be the exact Gibbs best response to $m^\star$.

The comparison below applies to any forward-consistent pair. For CENO, Theorem 1 gives the upper bound C_θ. Write

$$\epsilon = \mathrm{Expl}_\tau(\hat{\pi}, \hat{m}), \quad D = \sum_{h=0}^{H-1} \mathcal{M}_h(\hat{m}_h, m_h^\star). \tag{128}$$

Population and policy comparison. By the definition of exploitability,

$$\epsilon = \mathcal{J}_{\hat{m}}(\hat{\pi}) - \min_{\pi'} \mathcal{J}_{\hat{m}}(\pi') \geq \mathcal{J}_{\hat{m}}(\hat{\pi}) - \mathcal{J}_{\hat{m}}(\pi^\star). \tag{129}$$

Because the transition kernel does not depend on the population, evaluating a fixed policy under a different population field does not change that policy's state distribution. Expanding the two objectives, since $(\hat{\pi}, \hat{m})$ is forward-consistent, the state distribution generated by $\hat{\pi}_h$ is $\hat{m}_h$. Hence

$$\mathcal{J}_{\hat{m}}(\hat{\pi}) = \sum_{h=0}^{H-1} \sum_{x,a} \hat{m}_h(x)\hat{\pi}_h(a \mid x)[c_h(x,a) + f_h(x, \hat{m}_h) + \tau \log \hat{\pi}_h(a \mid x)] + \sum_x \hat{m}_H(x) g(x). \tag{130}$$

When the same policy $\hat{\pi}_h$ is evaluated under the population field $m^\star$, its state flow remains $\hat{m}$, because the transition kernel depends on the state, action, and time, but not on the population argument used in f_h. Therefore,

$$\mathcal{J}_{m^\star}(\hat{\pi}) = \sum_{h=0}^{H-1} \sum_{x,a} \hat{m}_h(x)\hat{\pi}_h(a \mid x)[c_h(x,a) + f_h(x, m_h^\star) + \tau \log \hat{\pi}_h(a \mid x)] + \sum_x \hat{m}_H(x) g(x). \tag{131}$$

Similarly, forward consistency of $(m^\star, \pi^\star)$ gives

$$\mathcal{J}_{m^\star}(\pi^\star) = \sum_{h=0}^{H-1} \sum_{x,a} m_h^\star(x)\pi_h^\star(a \mid x)[c_h(x,a) + f_h(x, m_h^\star) + \tau \log \pi_h^\star(a \mid x)] + \sum_x m_H^\star(x) g(x), \tag{132}$$

and

$$\mathcal{J}_{\hat{m}}(\pi^\star) = \sum_{h=0}^{H-1} \sum_{x,a} m_h^\star(x)\pi_h^\star(a \mid x)[c_h(x,a) + f_h(x, \hat{m}_h) + \tau \log \pi_h^\star(a \mid x)] + \sum_x m_H^\star(x) g(x). \tag{133}$$

Therefore,

$$\mathcal{J}_{\hat{m}}(\hat{\pi}) - \mathcal{J}_{m^\star}(\hat{\pi}) = \sum_{h=0}^{H-1} \sum_x \hat{m}_h(x)[f_h(x, \hat{m}_h) - f_h(x, m_h^\star)]. \tag{134}$$

and

$$\mathcal{J}_{\hat{m}}(\pi^\star) - \mathcal{J}_{m^\star}(\pi^\star) = \sum_{h=0}^{H-1} \sum_x m_h^\star(x)[f_h(x, \hat{m}_h) - f_h(x, m_h^\star)] \tag{135}$$

In each difference, the action costs, entropy terms, and terminal costs cancel because the policy and its state distribution is unchanged. Subtracting these identities yields

$$\mathcal{J}_{\hat{m}}(\hat{\pi}) - \mathcal{J}_{\hat{m}}(\pi^\star) = D + \mathcal{J}_{m^\star}(\hat{\pi}) - \mathcal{J}_{m^\star}(\pi^\star). \tag{136}$$

Consequently,

$$D + [\mathcal{J}_{m^\star}(\hat{\pi}) - \mathcal{J}_{m^\star}(\pi^\star)] \le \epsilon \tag{137}$$

This inequality does not use monotonicity. Since $\pi^\star$ is optimal under $m^\star$, the term in brackets is non-negative.

The policy cost gap under the equilibrium population. For each time step, define

$$K_h = \sum_x \hat{m}_h(x) \mathrm{KL}(\hat{\pi}_h(\cdot \mid x) \| \pi^\star_h(\cdot \mid x)). \tag{138}$$

Let $\phi^\star_h$ and $Q^\star_h$ be the exact soft Bellman values under $m^\star$, with $\phi^\star_H = g$. The Gibbs formula gives, at each state:

$$\sum_a \hat{\pi}_h(a \mid x)[Q^\star_h(x,a) + \tau \log \hat{\pi}_h(a \mid x)] - \phi^\star_h(x) = \tau \mathrm{KL}(\hat{\pi}_h(\cdot \mid x) \| \pi^\star_h(\cdot \mid x)). \tag{139}$$

After multiplying by $\hat{m}_h(x)$ and summing over states and time, the continuation-value terms cancel by forward consistency. Since the terminal value is g, this gives

$$\mathcal{J}_{m^\star}(\hat{\pi}) - \mathcal{J}_{m^\star}(\pi^\star) = \tau \sum_{h=0}^{H-1} K_h. \tag{140}$$

This is the policy-level KL identity applied to the fixed population $m^\star$. Combining it with the preceding comparison yields

$$D + \tau \sum_{h=0}^{H-1} K_h \le \epsilon. \tag{141}$$

The policy in this KL divergence is the equilibrium policy $\pi^\star$, which is a best response to $m^\star$.

Proof of Theorem 2.

Proof. Under ordinary monotonicity, $D \ge 0$. Therefore $\tau \sum_{h=0}^{H-1} K_h \le \epsilon$.

We relate the policy differences to the population differences. Define the state-action distributions

$$\hat{q}_h(x,a) = \hat{m}_h(x)\hat{\pi}_h(a \mid x), \quad q^\star_h(x,a) = m^\star_h(x)\pi^\star_h(a \mid x). \tag{142}$$

The chain rule for KL divergence gives

$$\mathrm{KL}(\hat{q}_h \| q^\star_h) = \mathrm{KL}(\hat{m}_h \| m^\star_h) + K_h. \tag{143}$$

Both state-action distributions are propagated by the same transition kernel $P_h(y \mid x, a)$. Applying the data-processing inequality therefore gives

$$\mathrm{KL}(\hat{m}_{h+1} \| m^\star_{h+1}) \le \mathrm{KL}(\hat{q}_h \| q^\star_h). \tag{144}$$

These are the standard chain-rule and common-kernel contraction properties of KL divergence.

Combining the two relations,

$$\mathrm{KL}(\hat{m}_{h+1} \| m^\star_{h+1}) \le \mathrm{KL}(\hat{m}_h \| m^\star_h) + K_h. \tag{145}$$

The relevant KL divergences are finite: the equilibrium Gibbs policy assigns positive probability to every admissible action, so any state reached by $\hat{\pi}$ from m_0 is also reached by $\pi^\star$.

Since $\hat{m}_0 = m_0 = m^\star_0$, induction gives, for $h = 1, \dots, H$,

$$\mathrm{KL}(\hat{m}_h \| m^\star_h) \le \sum_{t=0}^{h-1} K_t \le \frac{\epsilon}{\tau}. \tag{146}$$

Pinsker's inequality, with natural logarithms, gives

$$\|\hat{m}_h - m_h^\star\|_1^2 \le 2\mathrm{KL}(\hat{m}_h \| m_h^\star) \le \frac{2\epsilon}{\tau}. \tag{147}$$

The factor 2 here corresponds to the ℓ_1 distance, rather than total variation defined as half that distance. Meanwhile, the initial error is zero. Taking the maximum over all $h = 0, \dots, H$, and using Theorem 1 for the CENO output, gives

$$\max_{0 \le h \le H} \|\hat{m}_h - m_h^\star\|_1 \le \sqrt{\frac{2\epsilon}{\tau}} \le \sqrt{\frac{2C_\theta}{\tau}}. \tag{148}$$

This proves Theorem 2. □

Proof of Corollary 3. Under μ-strong monotonicity,

$$D \ge \mu \sum_{h=0}^{H-1} \|\hat{m}_h - m_h^\star\|_2^2. \tag{149}$$

Substituting this into the population and policy comparison gives the stronger joint estimate

$$\mu \sum_{h=0}^{H-1} \|\hat{m}_h - m_h^\star\|_2^2 + \tau \sum_{h=0}^{H-1} K_h \le \epsilon \le C_\theta. \tag{150}$$

Dropping the non-negative KL term proves Corollary 3:

$$\mu \sum_{h=0}^{H-1} \|\hat{m}_h - m_h^\star\|_2^2 \le \epsilon \le C_\theta. \tag{151}$$

In particular, the root-mean-square population error over the decision times satisfies

$$\left[\frac{1}{H} \sum_{h=0}^{H-1} \|\hat{m}_h - m_h^\star\|_2^2\right]^{1/2} \le \sqrt{\frac{C_\theta}{\mu H}}. \tag{152}$$

On the other hand, strong monotonicity also gives a full-trajectory ℓ_1 estimate through the finite-dimensional norm relation. For $h = 0, \dots, H-1$,

$$\|\hat{m}_h - m_h^\star\|_1 \le \sqrt{S}\|\hat{m}_h - m_h^\star\|_2 \le \sqrt{\frac{SC_\theta}{\mu}}, \quad S = |\mathcal{X}|. \tag{153}$$

To extend this estimate to H, we use CENO's terminal construction. Both CENO and equilibrium use the terminal value g, so their final-step state-action values satisfy

$$\hat{Q}_{H-1}(x,a) - Q_{H-1}^\star(x,a) = f_{H-1}(x, \hat{m}_{H-1}) - f_{H-1}(x, m_{H-1}^\star). \tag{154}$$

This difference is independent of the action. By the shift invariance of the Gibbs policy,

$$\hat{\pi}_{H-1}(\cdot \mid x) = \pi_{H-1}^\star(\cdot \mid x). \tag{155}$$

Thus the two terminal populations are obtained by applying the same Markov transition to their respective populations at $H-1$. The ℓ_1 non-expansion property gives

$$\|\hat{m}_H - m_H^\star\|_1 \le \|\hat{m}_{H-1} - m_{H-1}^\star\|_1. \tag{156}$$

Consequently, the strong-monotonicity estimate also holds at H.

Combining this estimate with Theorem 2 and the bound $\|m - m'\|_1 \le 2$ for probability distributions yields

$$\max_{0 \le h \le H} \|\hat{m}_h - m_h^\star\|_1 \le \min\left\{2, \sqrt{\frac{2C_\theta}{\tau}}, \sqrt{\frac{SC_\theta}{\mu}}\right\}. \tag{157}$$

The two estimates use different properties of the problem. The entropy-based estimate avoids the factor $\sqrt{S}$ while the strong-monotonicity estimate does not divide by τ.

Uniqueness of the equilibrium population. Ordinary monotonicity also gives uniqueness of the equilibrium population trajectory, whenever an equilibrium exists. Suppose (m^1, π^1) and (m^2, π^2) are two equilibria for the same initial population. Apply the preceding comparison with

$$(\hat{\pi}, \hat{m}) = (\pi^1, m^1), \quad (\pi^\star, m^\star) = (\pi^2, m^2). \tag{158}$$

Since the first pair is an equilibrium,

$$\epsilon = \mathrm{Expl}_\tau(\pi^1, m^1) = 0. \tag{159}$$

Monotonicity makes the interaction term non-negative, so

$$0 \le \tau \sum_{h=0}^{H-1} \sum_x m_h^1(x) \mathrm{KL}(\pi_h^1(\cdot \mid x) \| \pi_h^2(\cdot \mid x)) \le 0. \tag{160}$$

The population KL recursion used in the proof of Theorem 2 therefore gives

$$\mathrm{KL}(m_h^1 \| m_h^2) = 0, \quad h = 0, \dots, H. \tag{161}$$

Therefore,

$$m_h^1 = m_h^2, \quad h = 0, \dots, H. \tag{162}$$

Thus the equilibrium population trajectory is unique over the full horizon.

B Experimental Appendix

B.1 Beach Bar Experiment

Beach Bar Experiment Setting. The Beach Bar Problem is a classic model in game theory, also known as Hotelling's Spatial Competition Model, used to explain why competitors in a free market often choose virtually identical physical locations or product positions.

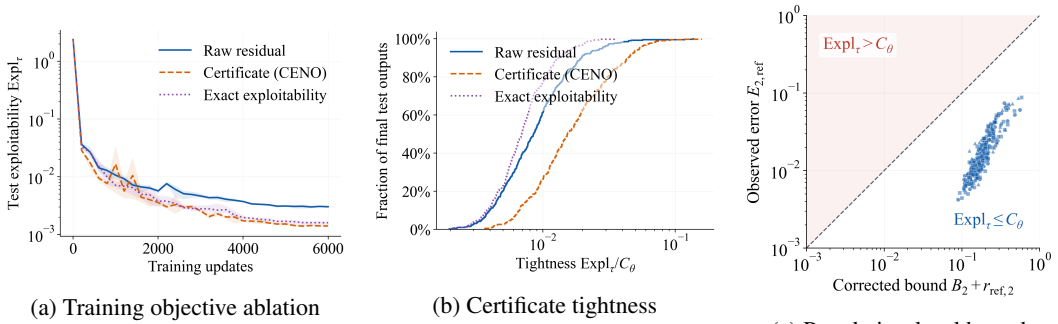
Here, we use a finite-horizon, entropy-regularized variant of this problem. The environment has 16×16 states, five actions, $\mathcal{A} = \{(-1, 0), (1, 0), (0, -1), (0, 1), (0, 0)\}$, and horizon $H = 12$. Transitions are deterministic and population-independent, with moves beyond the boundary remaining at the boundary. Agents prefer locations near the center but incur movement and congestion costs:

$$c_h(x, a) = 0.1|a| + 0.02|x_1 - 7.5| + 0.02|x_2 - 7.5|, \quad f_h(x, m_h) = \alpha m_h(x), \quad x = (x_1, x_2). \tag{163}$$

We set the terminal cost to zero and use a fixed entropy coefficient $\tau = 0.30$. Each instance $\xi = (m_0, \alpha)$ specifies an initial population distribution and a congestion coefficient, with $\alpha \sim \mathrm{Unif}[1, 4]$. We generate disjoint training, validation, and test sets of 1024, 128, and 128 instances from the same distribution.

All methods use the same experiment setting and objective. We measure true exploitability using an independent best-response computation against each policy's induced population trajectory. We plot exploitability against online wall-clock time and report CENO certificate alongside its actual exploitability. Online wall-clock time includes the complete construction of the policy and population trajectory, as well as certificate computation for CENO, but excludes offline training and independent exploitability evaluation.

Ablation study and population-level verification. Fig. 5a first compares the three training objectives under the same value-only parameterization. Certificate training reaches the lowest final test exploitability, followed closely by direct exploitability optimization, while raw Bellman residual converges to a higher error. This indicates that the certificate can serve as an effective label-free training signal. Fig. 5b further shows that certificate tightness depends on how the value function is trained. The CENO generally yields larger $\mathrm{Expl}_\tau / C_\theta$ ratios than raw residual model, whereas direct exploitability training gives smallest ratios. Finally, Fig. 5c compares the observed cumulative population error with the population bound. All tested outputs lie below the equality line, consistent with the population-level guarantee.



(a) Training objective ablation (b) Certificate tightness (c) Population-level bound

Figure 5: **Ablation study and population-level verification.** (a) Test exploitability during training for value-only models optimized with the raw Bellman residual, the CENO certificate, or exact exploitability. Curves show the mean over training seeds, with shaded regions indicating variation across seeds. (b) Empirical cumulative distributions of the tightness ratio $\text{Expl}_\tau/C_\theta$. $\text{Expl}_\tau/C_\theta$ closer to one indicate a tighter certificate, while smaller values correspond to more conservative bounds. (c) Population-level error versus the certificate-induced bound. The dashed diagonal denotes equality; points below it satisfy the bound after accounting for the numerical reference error.

B.2 Experimental Task Definition and Strong Monotonicity

Here, we provide the full task definition used in the experiments and verify that their population interactions satisfy the strong-monotonicity condition.

All tasks are defined on a 64×56 grid and use the action set

$$\mathcal{A} = \{\text{up, down, left, right, stay}\}. \tag{164}$$

For a traversable state x, let $T(x, a)$ denote the neighboring state reached by action a. A move outside the traversable region or into an obstacle remains at x. We use the deterministic transition

$$P_h(y \mid x, a) = \mathbf{1}\{y = T(x, a)\}, \tag{165}$$

which is independent of the population distribution. Target states are treated as absorbing states so that the total population is preserved throughout the finite horizon.

Each traversable cell has a capacity $C(x) = 8$. For an instance with N agents, $m_h(x)$ denotes the population mass at state x, and the corresponding expected cell count is

$$n_h(x) = N m_h(x), \tag{166}$$

and the local occupancy by

$$\rho_h(x) = \frac{n_h(x)}{C_h(x)}, \tag{167}$$

where $C_h(x) = 8$ unless the effective capacity is changed by the dynamic bottleneck in the Temporal Adaptation task.

The smooth congestion function is

$$B_\epsilon(\rho) = \frac{\bar{\rho}^2}{1 - \bar{\rho} + \epsilon}, \quad \bar{\rho} = \min\{\rho, 1 - \epsilon\}, \tag{168}$$

with $\epsilon = 0.05$. The interaction in the single-population tasks has the form

$$f_h(x, m) = \kappa m(x) + \beta B_\epsilon(\rho_h(x)), \quad \kappa > 0. \tag{169}$$

The first term gives a uniform response to changes in population mass, while the barrier term sharply increases the cost as local occupancy approaches its capacity.

Because B_ϵ is non-decreasing,

$$[m(x) - m'(x)][B_\epsilon(\rho(x)) - B_\epsilon(\rho'(x))] \geq 0. \tag{170}$$

Therefore,

$$\sum_x [m(x) - m'(x)][f_h(x, m) - f_h(x, m')]$$
$$\geq \kappa \sum_x [m(x) - m'(x)]^2 = \kappa \|m - m'\|_2^2. \tag{171}$$

Thus, the single-population interactions in the following tasks are strongly monotone with $\mu = \kappa > 0$.

To be specific, the multi-agent tasks use an open-system simulator with state-dependent batch releases, whereas each CENO planning call considers a finite-horizon MFG initialized from the currently observed population. At each environment step, the active population is normalized to define a new initial distribution, with capacity scaling adjusted consistently. Within the predicted rollout, no additional agents are injected and exited mass is retained in absorbing states. Only the first-step policy is executed, after which the observed state, including any new arrivals, defines the next planning instance.

For the evaluation metrics, we calculate the final exit rate by

$$R_{\text{exit}} = \frac{N_{\text{exited}}(T)}{N_{\text{exited}}(T) + N_{\text{active}}(T)}, \tag{172}$$

the corresponding barrier cost is

$$C_{\text{barrier}} = \sum_{t=0}^{T-1} \sum_x n_t(x) B_\epsilon(\rho_t(x)). \tag{173}$$

Spatial allocation. The first task uses a rectangular arena. The source area is placed at the top-left, while the exit window is on the right. Three obstacles separate four routes. The topmost and bottommost routes are only one cell wide; the upper route is shorter but substantially narrower than the lower one.

Let $d(x)$ denote the shortest-path distance from state x to the target. The individual task cost is

$$c_h(x, a) = d(T(x, a)) + \lambda_{\text{move}} 1_{a \neq \text{stay}} + \lambda_{\text{stay}} 1_{a=\text{stay}}, \tag{174}$$

with $\lambda_{\text{move}} = 0.01$, $\lambda_{\text{stay}} = 0.50$.

The population interaction is

$$f_h(x, m) = 0.10 m(x) + 8 B_{0.05}\left(\frac{N m(x)}{8}\right). \tag{175}$$

The task therefore trades a shorter travel distance against the higher congestion created when more population uses the narrow route.

Temporal adaptation. This task uses the same arena, source, target, wall, and geometry as the spatial allocation task. A hazard zone $\mathcal{H}$ covers the upper route and its neighborhood. The hazard becomes active during $h \in [50, 150)$.

Outside this interval, the effective capacity is 8; during the hazard interval, the effective capacity of states in hazard zone is reduced to 4, while the route remains traversable. The environment also uses a hazard penalty.

Define

$$C_h(x) = \begin{cases} 4, & x \in \mathcal{H}, \quad 50 \leq h < 150, \\ 8, & \text{otherwise}, \end{cases} \tag{176}$$

and the penalty:

$$q_h(x) = 25\, \mathbf{1}_{x \in \mathcal{H}} \mathbf{1}_{50 \leq h < 150}. \tag{177}$$

The task cost is

$$c_h(x, a) = d(T(x, a)) + \lambda_{\text{move}} 1_{a \neq \text{stay}} + \lambda_{\text{stay}} 1_{a=\text{stay}} + q_h(T(x, a)), \tag{178}$$

with $\lambda_{\text{move}} = 0.01$, $\lambda_{\text{stay}} = 0.50$. And the population interaction is

$$f_h(x, m) = 0.10m(x) + 8B_{0.05}\left(\frac{Nm(x)}{C_h(x)}\right). \tag{179}$$

The time dependence enters through the hazard schedule and the effective capacity $C_h(x)$, but not through the transition kernel. Since $C_h(x) > 0$ for every state and time and B_ϵ remains non-decreasing, the same argument gives

$$\sum_x (m - m')(f_h(m) - f_h(m')) \geq \kappa \|m - m'\|_2^2. \tag{180}$$

Thus, this task is also strongly monotone uniformly over time.

Counterflow coordination. This task uses a shared rectangular arena. Each group starts from an opposite area and moves in opposite directions to the corresponding exit zone. We represent this setting within a single finite-state formulation by augmenting the state with the population type,

$$\tilde{\mathcal{X}} = \mathcal{X} \times \{A, B\}. \tag{181}$$

The transition preserves the population type. Let $m^A(x)$ and $m^B(x)$ denote the two local population masses, and let

$$s(x) = m^A(x) + m^B(x). \tag{182}$$

For each population $p \in \{A, B\}$, the individual task cost has the same form:

$$c_h^p(x, a) = d_p(T(x, a)) + \lambda_{\text{move}} 1_{a \neq \text{stay}} + \lambda_{\text{stay}} 1_{a = \text{stay}}, \tag{183}$$

with $\lambda_{\text{move}} = 0.01$, $\lambda_{\text{stay}} = 0.50$. The interaction is defined by

$$\begin{aligned} f^A(x, m) &= \kappa m^A(x) + \gamma m^B(x) + 6B_{0.05}\left(\frac{Ns(x)}{8}\right), \\ f^B(x, m) &= \kappa m^B(x) + \gamma m^A(x) + 6B_{0.05}\left(\frac{Ns(x)}{8}\right), \end{aligned} \tag{184}$$

where $\gamma > 0$ penalizes overlap with the opposite population. To ensure strong monotonicity, we need to keep $\kappa > \gamma$. Here, we choose $\kappa = 0.15, \gamma = 0.05$.

Proof. To verify strong monotonicity, let

$$z_A = m^A - m^{A'}, \quad z_B = m^B - m^{B'}. \tag{185}$$

The linear part of the interaction in Appendix A.6 gives, at each state,

$$\kappa(z_A^2 + z_B^2) + 2\gamma z_A z_B. \tag{186}$$

Its associated matrix is

$$\begin{bmatrix} \kappa & \gamma \\ \gamma & \kappa \end{bmatrix} \tag{187}$$

whose eigenvalues are $\kappa + \gamma$ and $\kappa - \gamma$. Hence,

$$\kappa(z_A^2 + z_B^2) + 2\gamma z_A z_B \geq (\kappa - \gamma)(z_A^2 + z_B^2). \tag{188}$$

The barrier term, which depends only on the total population $s(x)$, is non-decreasing, so its contribution to the monotonicity inner product is non-negative. Therefore,

$$\langle m - m', f(m) - f(m') \rangle \geq (\kappa - \gamma)\|m - m'\|_2^2. \tag{189}$$

Thus, the counterflow task is strongly monotone with

$$\mu = \kappa - \gamma > 0. \tag{190}$$

□

In fact, the interaction function in Eq.(184) can be extended to two completely heterogeneous groups:

$$f_{\text{lin}}^{A} = \kappa_A m^A + \gamma_{AB} m^B + b(s),$$
$$f_{\text{lin}}^{B} = \kappa_B m^B + \gamma_{BA} m^A + b(s), \tag{191}$$

the linear part becomes

$$\kappa_A z_A^2 + \kappa_B z_B^2 + (\gamma_{AB} + \gamma_{BA}) z_A z_B, \tag{192}$$

the corresponding symmetric matrix is

$$\begin{bmatrix} \kappa_A & \frac{\gamma_{AB}+\gamma_{BA}}{2} \\ \frac{\gamma_{AB}+\gamma_{BA}}{2} & \kappa_B \end{bmatrix} \tag{193}$$

the only requirement is positive definiteness:

$$4\kappa_A \kappa_B > (\gamma_{AB} + \gamma_{BA})^2, \quad \kappa_A > 0, \kappa_B > 0. \tag{194}$$

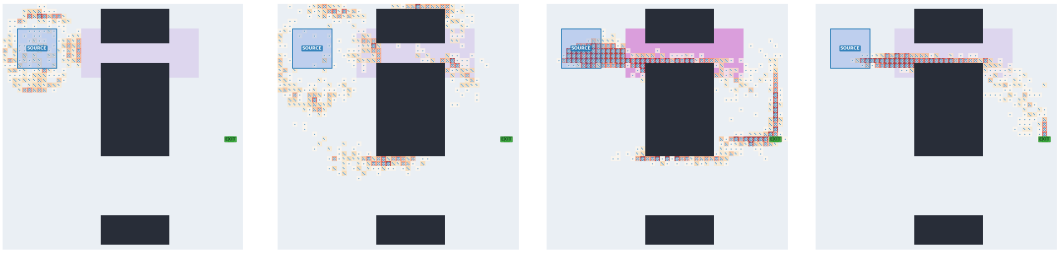
Here, we can take

$$\mu = \lambda_{\min} = \frac{\kappa_A + \kappa_B - \sqrt{(\kappa_A - \kappa_B)^2 + (\gamma_{AB} + \gamma_{BA})^2}}{2} \tag{195}$$

The condition in our experiment is the special one:

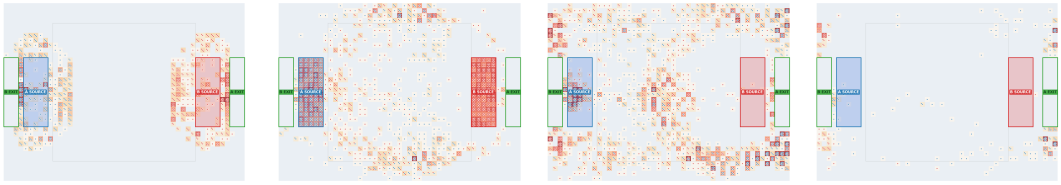
$$\kappa_A = \kappa_B = \kappa, \gamma_{AB} = \gamma_{BA} = \gamma \Rightarrow \mu = \kappa - \gamma. \tag{196}$$

B.3 Multi-agent Task Results



(a) $t = 5$ (b) $t = 40$ (c) $t = 130$ (d) $t = 240$

Figure 6: **Population evolution in the temporal adaptation task.** Snapshots show how the population evolves before, during, and after the hazard. The purple region denotes the future/active soft bottleneck, whose effective capacity is reduced during the hazard interval.



(a) $t = 5$ (b) $t = 35$ (c) $t = 125$ (d) $t = 240$

Figure 7: Population evolution in the counterflow coordination task. Snapshots show two populations moving in opposite directions through the shared space. Agents disperse across the entire space to avoid congestion in the central area.

Fig. 6 illustrates how CENO responds to the bottleneck. Before the hazard becomes active, the population already spreads across alternative passages rather than committing entirely to the affected upper route. During the active period ($t = 130$), agents continue to use both routes, while avoiding excessive concentration inside the reduced-capacity region. After the bottleneck is removed, the upper route becomes less dense, so agents tend to move through it. This evolution complements

the quantitative results by showing that CENO adapts the population trajectory to a future time-dependent change rather than reacting only after it becomes active.

Fig. 7 shows the evolution of the two opposing populations. Starting from opposite source regions, the two groups spread through the shared space and use multiple portions of the corridor instead of forming a single highly concentrated opposing stream. As the rollout progresses, agents from both populations continue to cross the shared region while maintaining movement toward their respective exits.

B.4 Implementation Details

All experiments are implemented in PyTorch 2.8 and conducted on a workstation with an Intel Core i9-13900K CPU, an NVIDIA GeForce RTX 4090 GPU, and 64 GB of system memory.

Since the forward consistency is structurally ensured, we build an end-to-end structure that gradients are propagated end-to-end through the policy construction and the population dynamics. For CENO, we use a Fourier Neural Operator (FNO) as the value-only neural part across all experiments. First, we use a MLP encoder to unify each instance, then the FNO uses a hidden width of 128, four spectral layers, and 12 Fourier modes along each spatial dimension, with a padding size of 6. Normalized spatial coordinates are appended to the input, and each spectral layer combines a Fourier convolution with a pointwise convolution followed by a GELU activation. For the state-wise maximum in C_h, we use the standard subgradient of the exact max operator. The network predicts the continuation values over the planning horizon in one forward pass, while the policy and population trajectory are subsequently constructed through the Gibbs policy and the policy-induced Markov rollout. Since the network is evaluated once per instance, the total cost is $\mathcal{O}(HSA)$ and no fixed-point loop. For all baselines, we use the official implementations. We adapt only the observation/action interface and task reward to the common grid environments. All codes and results are attached in the supplementary materials and will be open to public after review.