A Unified Framework for Locality in Scalable MARL

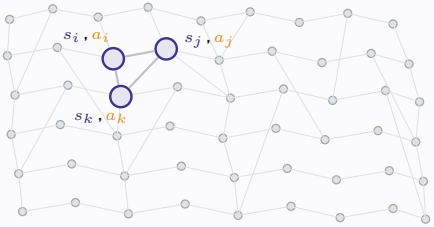
Sourav Chakraborty* Amit Kiran Rege* Claire Monteleoni Lijun Chen

University of Colorado Boulder · INRIA Paris *equal contribution

The scalability problem

n agents with coupled dynamics — joint spaces explode

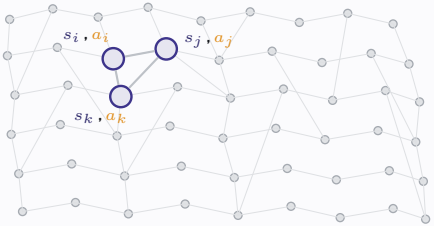
- n agents on a network. Each has local state $s_i \in \mathcal{S}_i$ and action $a_i \in \mathcal{A}_i$.



The scalability problem

n agents with coupled dynamics — joint spaces explode

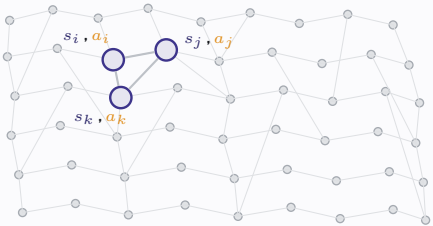
- n agents on a network. Each has local state $s_i \in \mathcal{S}_i$ and action $a_i \in \mathcal{A}_i$.
- **Centralized RL** must act on the joint $(s, a) = (s_1, \ldots, s_n, a_1, \ldots, a_n)$.



The scalability problem

n agents with coupled dynamics — joint spaces explode

- n agents on a network. Each has local state $s_i \in \mathcal{S}_i$ and action $a_i \in \mathcal{A}_i$.
- **Centralized RL** must act on the joint $(s, a) = (s_1, \ldots, s_n, a_1, \ldots, a_n)$.
- Both spaces blow up: $|\mathcal{S}| = |\mathcal{S}_i|^n$, $|\mathcal{A}| = |\mathcal{A}_i|^n$ — **exponential** in n.



Centralized RL is hopeless. Scalable MARL must exploit structure — **but what structure, and how?**

The escape: locality

Influence decays exponentially with graph distance (EDP)

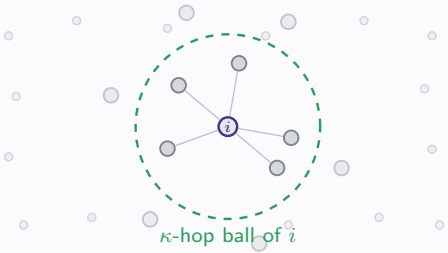
- **Key insight** (Qu, Lin, Wierman, Li): agent j's influence on agent i's value decays exponentially in their graph distance.



The escape: locality

Influence decays exponentially with graph distance (EDP)

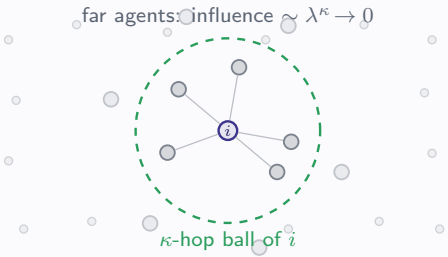
- **Key insight** (Qu, Lin, Wierman, Li): agent j's influence on agent i's value decays exponentially in their graph distance.
- **Exponential Decay Property (EDP)** $\Rightarrow$ truncate to a **κ-hop neighborhood**.



The escape: locality

Influence decays exponentially with graph distance (EDP)

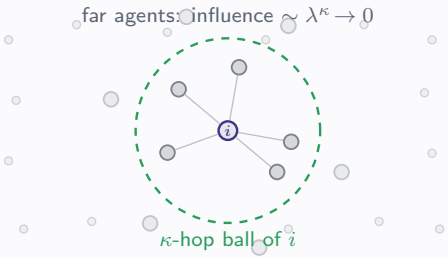
- **Key insight** (Qu, Lin, Wierman, Li): agent j's influence on agent i's value decays exponentially in their graph distance.
- **Exponential Decay Property (EDP)** $\Rightarrow$ truncate to a κ-**hop neighborhood**.
- Complexity scales with **neighborhood size**, not n.



The escape: locality

Influence decays exponentially with graph distance (EDP)

- **Key insight** (Qu, Lin, Wierman, Li): agent j's influence on agent i's value decays exponentially in their graph distance.
- **Exponential Decay Property (EDP)** $\Rightarrow$ truncate to a κ-**hop neighborhood**.
- Complexity scales with **neighborhood size**, not n.

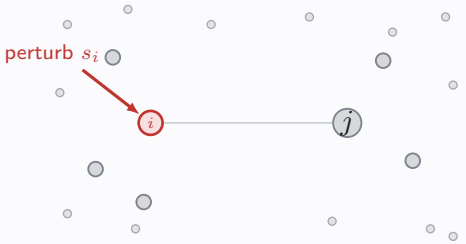


EDP justifies κ-hop truncation. **But when does EDP actually hold?** That's what this paper answers.

What couples two agents?

The coupling coefficient $C_{j \leftarrow i}$ — the unit of influence

- Perturb agent i's state. How much does agent j's *next state* move?



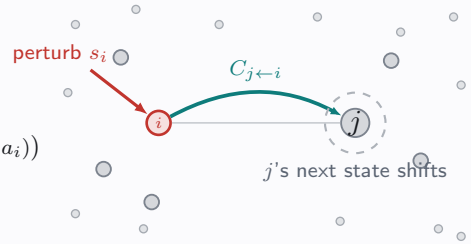
What couples two agents?

The coupling coefficient $C_{j\leftarrow i}$ — the unit of influence

- Perturb agent i's state. How much does agent j's *next state* move?
- The **coupling coefficient** (prior work):

$$C_{j\leftarrow i} = \sup_{s_i,\, s_i',\, a_i} \mathrm{TV}\big(P_j(\cdot \mid s_i, s_{-i}, a_i),\, P_j(\cdot \mid s_i', s_{-i}, a_i)\big)$$

Worst case over i's state *and* action, all else held fixed. Stack into a matrix C: entry (j, i) is $C_{j\leftarrow i}$



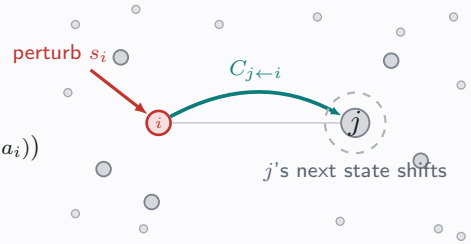
What couples two agents?

The coupling coefficient $C_{j\leftarrow i}$ — the unit of influence

- Perturb agent i's state. How much does agent j's *next state* move?
- The **coupling coefficient** (prior work):

$$C_{j\leftarrow i} = \sup_{s_i, s_i', a_i} \mathrm{TV}\big(P_j(\cdot \mid s_i, s_{-i}, a_i),\, P_j(\cdot \mid s_i', s_{-i}, a_i)\big)$$

Worst case over i's state *and* action, all else held fixed. Stack into a matrix C: entry (j, i) is $C_{j\leftarrow i}$



It's the **strongest possible** jolt that a change at i can deliver to j, regardless of what the policy actually does.

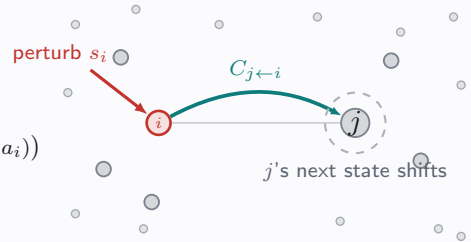
What couples two agents?

The coupling coefficient $C_{j\leftarrow i}$ — the unit of influence

- Perturb agent i's state. How much does agent j's *next state* move?
- The **coupling coefficient** (prior work):

$$C_{j\leftarrow i} = \sup_{s_i, s_i', a_i} \mathrm{TV}\big(P_j(\cdot \mid s_i, s_{-i}, a_i),\, P_j(\cdot \mid s_i', s_{-i}, a_i)\big)$$

Worst case over i's state *and* action, all else held fixed. Stack into a matrix C: entry (j, i) is $C_{j\leftarrow i}$

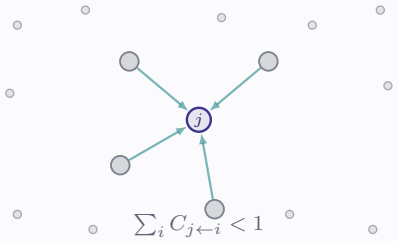


It's the **strongest possible** jolt that a change at i can deliver to j, regardless of what the policy actually does.

The prior answer: $\|C\|_\infty < 1$

Dobrushin weak-coupling — sufficient for EDP to hold

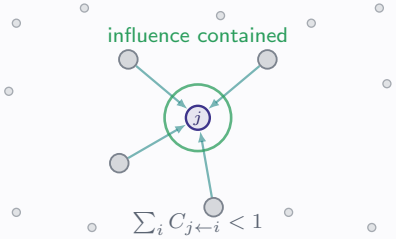
- Prior work (Qu, Lin, Wierman, Li; 2019–2021): if the row-sum norm $\|C\|_\infty = \max_j \sum_i C_{j\leftarrow i} < 1$, then EDP holds with $\lambda = \|C\|_\infty$.



The prior answer: $\|C\|_\infty < 1$

Dobrushin weak-coupling — sufficient for EDP to hold

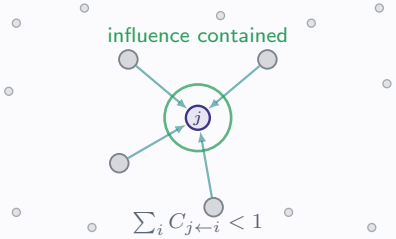
- Prior work (Qu, Lin, Wierman, Li; 2019–2021): if the row-sum norm $\|C\|_\infty = \max_j \sum_i C_{j \leftarrow i} < 1$, then EDP holds with $\lambda = \|C\|_\infty$.
- Each agent j is **weakly influenced in aggregate** by all its neighbours



The prior answer: $\|C\|_\infty < 1$

Dobrushin weak-coupling — sufficient for EDP to hold

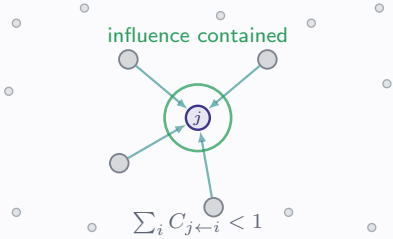
- Prior work (Qu, Lin, Wierman, Li; 2019–2021): if the row-sum norm $\|C\|_\infty = \max_j \sum_i C_{j\leftarrow i} < 1$, then EDP holds with $\lambda = \|C\|_\infty$.
- Each agent j is **weakly influenced in aggregate** by all its neighbours
- **The catch**: $C_{j\leftarrow i} = \sup_a(\cdots)$ — worst case over all actions. The **policy is invisible** to this condition.



The prior answer: $\|C\|_\infty < 1$

Dobrushin weak-coupling — sufficient for EDP to hold

- Prior work (Qu, Lin, Wierman, Li; 2019–2021): if the row-sum norm $\|C\|_\infty = \max_j \sum_i C_{j\leftarrow i} < 1$, then EDP holds with $\lambda = \|C\|_\infty$.
- Each agent j is **weakly influenced in aggregate** by all its neighbours
- **The catch**: $C_{j\leftarrow i} = \sup_a(\cdots)$ — worst case over all actions. The **policy is invisible** to this condition.



$C_{j\leftarrow i} = \sup_a(\cdots)$
worst case —
policy invisible

The limitation: a policy-independent condition

Prior EDP condition conflates environment and policy

- Prior EDP condition: $\|C\|_\infty < 1$, $C_{j \leftarrow i} = \sup_a(\cdots)$.
- Supremum over **all actions**: could some action ever couple i and j?



The limitation: a policy-independent condition

Prior EDP condition conflates environment and policy

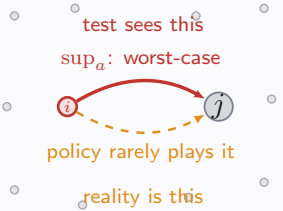
- Prior EDP condition: $\|C\|_\infty < 1$, $C_{j \leftarrow i} = \sup_a(\cdots)$.
- Supremum over **all actions**: could some action ever couple i and j?
- But the policy may **never play** those actions, or may be **insensitive** to s_i entirely.



The limitation: a policy-independent condition

Prior EDP condition conflates environment and policy

- Prior EDP condition: $\|C\|_\infty < 1$, $C_{j \leftarrow i} = \sup_a(\cdots)$.
- Supremum over **all actions**: could some action ever couple i and j?
- But the policy may **never play** those actions, or may be **insensitive** to s_i entirely.
- The test sees **worst-case physics**. It is blind to the actual **logic** of the policy.



A **worst-case property of the environment** cannot certify locality when the **policy** quietly breaks the coupling.

Splitting the coupling: physics vs. logic

Three matrices — only one depends on the policy

How does $C^{\pi}_{j \leftarrow i}$ arise? Two routes, three matrices:

- $E^{s}_{j \leftarrow i}$ — state sensitivity of environment:

$$\sup_{\substack{s, s' : s_{-i} = s'_{-i} \\ a \in \mathcal{A}}} \mathrm{TV}(P_j(\cdot|s,a),\, P_j(\cdot|s',a))$$

  s_i *changes, action fixed — pure environment.*

E^{s}

state $\rightarrow$ next state

environment only

Splitting the coupling: physics vs. logic

Three matrices — only one depends on the policy

How does $C^{\pi}_{j \leftarrow i}$ arise? Two routes, three matrices:

- $E^{s}_{j \leftarrow i}$ — state sensitivity of environment:

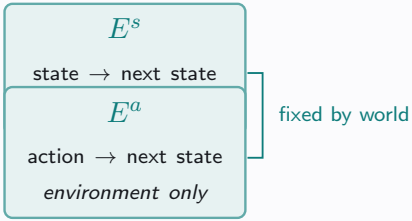
$$\sup_{\substack{s,s' \colon s_{-i} = s'_{-i} \\ a \in \mathcal{A}}} \mathrm{TV}(P_j(\cdot|s,a),\, P_j(\cdot|s',a))$$

  s_i changes, action fixed — pure environment.

- $E^{a}_{j \leftarrow k}$ — action sensitivity of environment:

$$\sup_{\substack{s \in \mathcal{S} \\ a,a' \colon a_{-k} = a'_{-k}}} \mathrm{TV}(P_j(\cdot|s,a),\, P_j(\cdot|s,a'))$$

  action k changes, state fixed — pure environment.

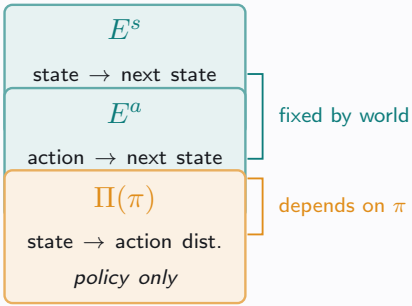


Splitting the coupling: physics vs. logic

Three matrices — only one depends on the policy

How does $C^{\pi}_{j \leftarrow i}$ arise? Two routes, three matrices:

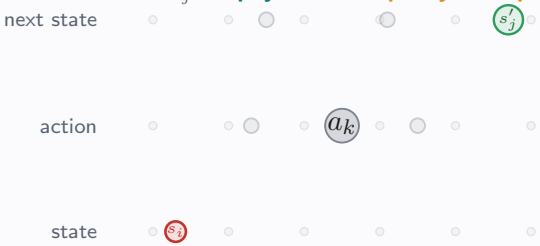
- $E^s_{j \leftarrow i}$ — state sensitivity of environment:
  $$\sup_{\substack{s,s' \colon s_{-i} = s'_{-i} \\ a \in \mathcal{A}}} \mathrm{TV}(P_j(\cdot|s,a),\, P_j(\cdot|s',a))$$
  s_i changes, action fixed — pure environment.
- $E^a_{j \leftarrow k}$ — action sensitivity of environment:
  $$\sup_{\substack{s \in \mathcal{S} \\ a,a' \colon a_{-k} = a'_{-k}}} \mathrm{TV}(P_j(\cdot|s,a),\, P_j(\cdot|s,a'))$$
  action k changes, state fixed — pure environment.
- $\Pi_{k \leftarrow i}(\pi)$ — state sensitivity of policy:
  $$\sup_{s,s' \colon s_{-i} = s'_{-i}} \mathrm{TV}(\pi_k(\cdot|s_{O_k}),\, \pi_k(\cdot|s'_{O_k}))$$
  s_i changes — how much does agent k's action shift?



E^s, E^a are **fixed properties of the environment**. $\Pi(\pi)$ is the **only piece the policy controls**.

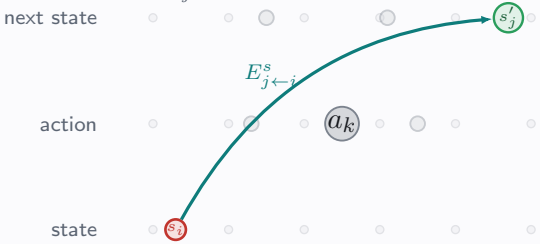
Two routes a disturbance can travel — and the equation

One poke at s_i reaches s'_j via **physics** *or via* **policy then physics**



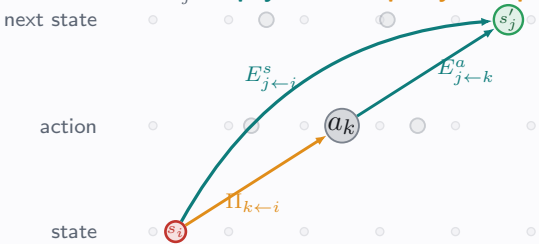
Two routes a disturbance can travel — and the equation

One poke at s_i reaches s'_j via **physics** *or via* **policy then physics**



Two routes a disturbance can travel — and the equation

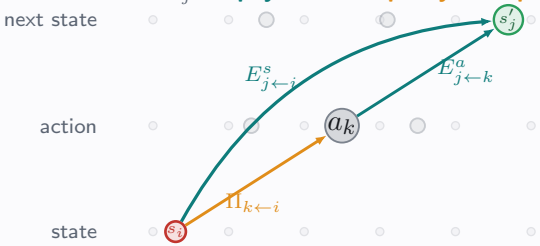
One poke at s_i reaches s'_j via **physics** *or via* **policy then physics**



sum over k = matrix product $E^a\Pi$

Two routes a disturbance can travel — and the equation

One poke at s_i reaches s'_j via **physics** *or via* **policy then physics**

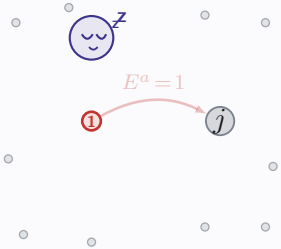


$$C^\pi \preceq \underbrace{E^s}_{\text{physics}} + \underbrace{E^a\Pi(\pi)}_{\text{logic}}$$

E^s, E^a are **fixed by the world**. $\Pi(\pi)$ is the **only policy-dependent piece**. (Prop. 1)

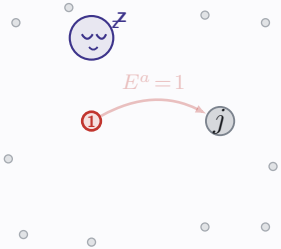
Small Example

- $E^s = 0$ (state: no direct effect)
- $E^a_{j \leftarrow 1} = 1$ (**maximal** action coupling)



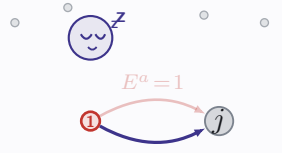
Small Example

- $E^s = 0$ (state: no direct effect)
- $E^a_{j \leftarrow 1} = 1$ (**maximal** action coupling)
- smooth policy: $\Pi_{1 \leftarrow 1}(\pi) = \alpha \ll 1$



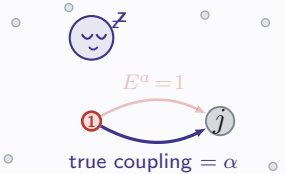
Small Example

- $E^s = 0$ (state: no direct effect)
- $E^a_{j \leftarrow 1} = 1$ (**maximal** action coupling)
- smooth policy: $\Pi_{1 \leftarrow 1}(\pi) = \alpha \ll 1$
- $C^\pi_{j \leftarrow 1} \leq \underbrace{0}_{E^s} + \underbrace{1}_{E^a} \cdot \underbrace{\alpha}_{\Pi} = \alpha$



Small Example

- $E^s = 0$ (state: no direct effect)
- $E^a_{j \leftarrow 1} = 1$ (**maximal** action coupling)
- smooth policy: $\Pi_{1 \leftarrow 1}(\pi) = \alpha \ll 1$
- $C^\pi_{j \leftarrow 1} \leq \underbrace{0}_{E^s} + \underbrace{1}_{E^a} \cdot \underbrace{\alpha}_{\Pi} = \alpha$

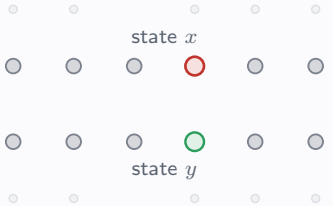


The same "1" the test choked on is now **tamed to** α by the closed loop.

The decomposition changes the verdict. **Policy logic** *does* determine locality.

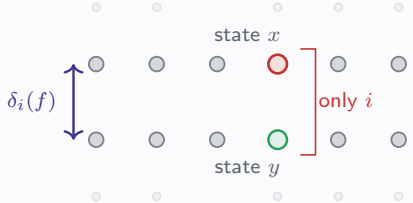
The measuring ruler: coordinatewise oscillation

- Wiggle *only* coordinate i; hold everything else fixed. How much can f change?



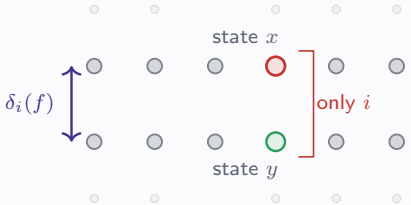
The measuring ruler: coordinatewise oscillation

- Wiggle *only* coordinate i; hold everything else fixed. How much can f change?
- $\delta_i(f) = \sup_{x,y:\, x_{-i}=y_{-i}} |f(x) - f(y)|$



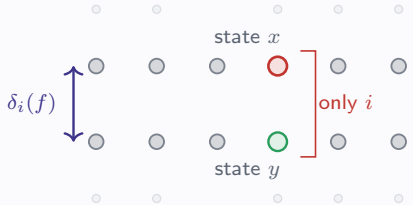
The measuring ruler: coordinatewise oscillation

- Wiggle *only* coordinate i; hold everything else fixed. How much can f change?
- $\delta_i(f) = \sup_{x,y:\, x_{-i}=y_{-i}} |f(x) - f(y)|$
- The **Dobrushin influence coefficient** — single-coordinate Lipschitz constant.



The measuring ruler: coordinatewise oscillation

- Wiggle *only* coordinate i; hold everything else fixed. How much can f change?
- $\delta_i(f) = \sup_{x,y:\, x_{-i}=y_{-i}} |f(x) - f(y)|$
- The **Dobrushin influence coefficient** — single-coordinate Lipschitz constant.
- Stack into a vector $\delta(f) = (\delta_i(f))_i$. Everything is phrased in this ruler.



The object we want to bound: differential value h^π

- The **Bellman operator** T^π acts on any bounded $f : \mathcal{S} \to \mathbb{R}$ by

$$T^\pi f(s) \;=\; \mathbb{E}[f(S_{t+1}) \mid S_t = s],$$

the one-step expectation under the synchronous closed-loop kernel.

The object we want to bound: differential value h^π

- The **Bellman operator** T^π acts on any bounded $f : \mathcal{S} \to \mathbb{R}$ by

$$T^\pi f(s) \;=\; \mathbb{E}[f(S_{t+1}) \mid S_t = s],$$

  the one-step expectation under the synchronous closed-loop kernel.

- The **differential value function** h^π is the solution to the **average-reward Poisson equation**:

$$h^\pi(s) \;=\; r^\pi(s) - \bar{r}^\pi \;+\; \mathbb{E}[h^\pi(S_{t+1}) \mid S_t = s],$$

  or in operator form, $\quad h^\pi - T^\pi h^\pi \;=\; r^\pi - \bar{r}^\pi$.

  $h^\pi(s)$ is the long-run *relative* value of state s above average.

The object we want to bound: differential value h^π

- The **Bellman operator** T^π acts on any bounded $f : \mathcal{S} \to \mathbb{R}$ by

$$T^\pi f(s) \;=\; \mathbb{E}[f(S_{t+1}) \mid S_t = s],$$

  the one-step expectation under the synchronous closed-loop kernel.

- The **differential value function** h^π is the solution to the **average-reward Poisson equation**:

$$h^\pi(s) \;=\; r^\pi(s) - \bar{r}^\pi \;+\; \mathbb{E}[h^\pi(S_{t+1}) \mid S_t = s],$$

  or in operator form, $\quad h^\pi - T^\pi h^\pi \;=\; r^\pi - \bar{r}^\pi$.

  $h^\pi(s)$ is the long-run *relative* value of state s above average.

- **Subtlety**: there is no $\gamma < 1$, so T^π is **not** a contraction in $\|\cdot\|_\infty$. Adding any constant to h^π gives another valid solution — h^π is unique only **up to an additive constant**.

The object we want to bound: differential value h^π

- The **Bellman operator** T^π acts on any bounded $f : \mathcal{S} \to \mathbb{R}$ by

$$T^\pi f(s) \;=\; \mathbb{E}[f(S_{t+1}) \mid S_t = s],$$

  the one-step expectation under the synchronous closed-loop kernel.

- The **differential value function** h^π is the solution to the **average-reward Poisson equation**:

$$h^\pi(s) \;=\; r^\pi(s) - \bar{r}^\pi \;+\; \mathbb{E}[h^\pi(S_{t+1}) \mid S_t = s],$$

  or in operator form, $\quad h^\pi - T^\pi h^\pi \;=\; r^\pi - \bar{r}^\pi$.

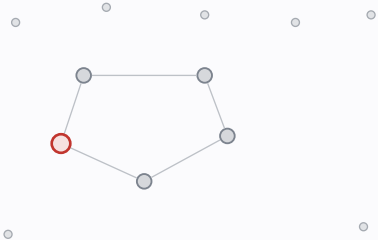
  $h^\pi(s)$ is the long-run *relative* value of state s above average.

- **Subtlety**: there is no $\gamma < 1$, so T^π is **not** a contraction in $\|\cdot\|_\infty$. Adding any constant to h^π gives another valid solution — h^π is unique only **up to an additive constant**.

- **Fix**: the oscillation seminorm $\delta_i(\cdot)$ is **blind to constants** by definition. It quotients out exactly the flat direction where T^π fails to contract.

One step of the loop maps oscillations by H^π

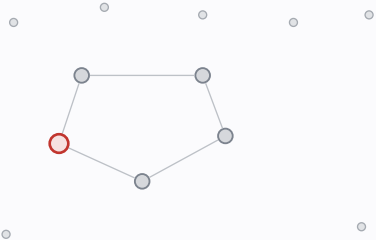
- Define $H^\pi := E^s + E^a \Pi(\pi)$.



One step of the loop maps oscillations by H^π

- Define $H^\pi := E^s + E^a \Pi(\pi)$.
- One step of the closed-loop kernel:

$$\delta(T^\pi f) \;\leq\; (H^\pi)^\top \delta(f).$$

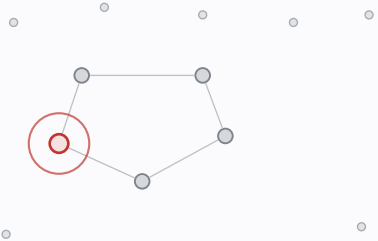


One step of the loop maps oscillations by H^π

- Define $H^\pi := E^s + E^a \Pi(\pi)$.
- One step of the closed-loop kernel:

$$\delta(T^\pi f) \;\leq\; (H^\pi)^\top \delta(f).$$

- A poke at i becomes H^π applied to that poke. Iterating gives $(H^\pi)^t$. (Lemma 2)



One step of the loop maps oscillations by H^π

- Define $H^\pi := E^s + E^a \Pi(\pi)$.
- One step of the closed-loop kernel:

$$\delta(T^\pi f) \;\le\; (H^\pi)^\top \delta(f).$$

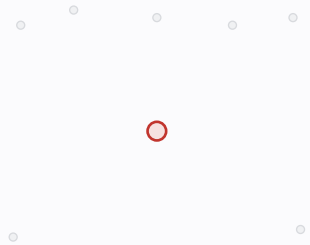
- A poke at i becomes H^π applied to that poke. Iterating gives $(H^\pi)^t$. (Lemma 2)
- If $\rho(H^\pi) < 1$, powers decay $\Rightarrow$ **spectral radius decides everything**.



The spectral condition: $\rho(H^\pi) < 1$

Poisson equation solvable — and sensitivity is bounded

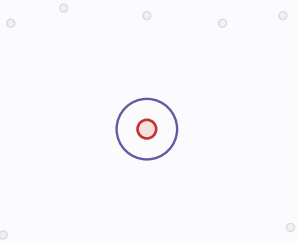
- $\rho(H^\pi) \leq \lambda < 1 \Rightarrow$ the **Poisson equation** is solvable (unique up to constant).



The spectral condition: $\rho(H^\pi) < 1$

Poisson equation solvable — and sensitivity is bounded

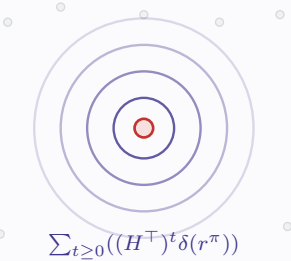
- $\rho(H^\pi) \leq \lambda < 1 \Rightarrow$ the **Poisson equation** is solvable (unique up to constant).
- $\delta(h^\pi) \leq \left(I - (H^\pi)^\top\right)^{-1} \delta(r^\pi)$



The spectral condition: $\rho(H^\pi) < 1$

Poisson equation solvable — and sensitivity is bounded

- $\rho(H^\pi) \leq \lambda < 1 \Rightarrow$ the **Poisson equation** is solvable (unique up to constant).
- $\delta(h^\pi) \leq \left(I - (H^\pi)^\top\right)^{-1} \delta(r^\pi)$
- The resolvent = **sum over all path lengths**, each damped by ρ: $\sum_{t \geq 0} ((H^\pi)^\top)^t \delta(r^\pi)$.

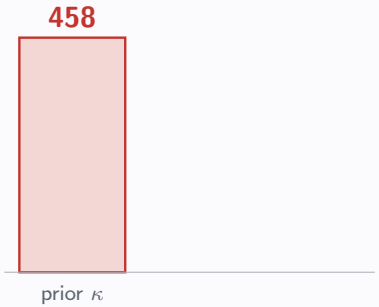


Value sensitivity, **propagated over every path**, damped by $\rho(H^\pi)$. (Theorem 3)

The payoff: Comparison with the Discounted Case.

$\gamma = 0.99, \quad \rho(H^\pi) = 0.5,$ target error 0.01.

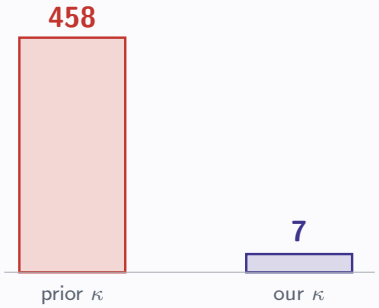
- **Prior:** $\lambda = \gamma = 0.99$
  $\Rightarrow \kappa \approx \mathbf{458}$ hops.



The payoff: Comparison with the Discounted Case.

$\gamma = 0.99, \quad \rho(H^\pi) = 0.5, \quad$ target error 0.01.

- **Prior:** $\lambda = \gamma = 0.99$
  $\Rightarrow \kappa \approx \mathbf{458}$ hops.
- **Ours:** $\lambda = 0.99 \times 0.5 = 0.495$
  $\Rightarrow \kappa \approx \mathbf{7}$ hops.

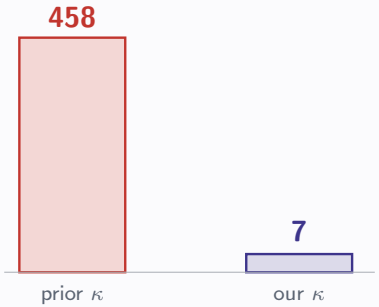


The payoff: Comparison with the Discounted Case.

$\gamma = 0.99, \quad \rho(H^\pi) = 0.5,$ target error 0.01.

- **Prior:** $\lambda = \gamma = 0.99$
  $\Rightarrow \kappa \approx \mathbf{458}$ hops.
- **Ours:** $\lambda = 0.99 \times 0.5 = 0.495$
  $\Rightarrow \kappa \approx \mathbf{7}$ hops.

Same guarantee. **Two orders of magnitude** less communication.



Temperature τ is a provable locality knob

- A **softmax policy** at temperature τ:

$$\pi_k(a_k \mid s_{O_k}) \;\propto\; \exp\!\big(g_k(s_{O_k}, a_k)/\tau\big).$$

Higher τ = more uniform over actions; lower τ = more concentrated (greedy).

Temperature τ is a provable locality knob

- A **softmax policy** at temperature τ:

$$\pi_k(a_k \mid s_{O_k}) \;\propto\; \exp\big(g_k(s_{O_k}, a_k)/\tau\big).$$

  Higher τ = more uniform over actions; lower τ = more concentrated (greedy).

- As τ grows, the policy reacts less to its state input, so the policy sensitivity matrix vanishes:

$$\tau \to \infty \implies \Pi(\pi) \to 0 \implies H^\pi = E^s + E^a \Pi(\pi) \to E^s.$$

  If the environment alone is stable ($\rho(E^s) < 1$), high entropy **guarantees** locality.

Temperature τ is a provable locality knob

- A **softmax policy** at temperature τ:

$$\pi_k(a_k \mid s_{O_k}) \;\propto\; \exp\big(g_k(s_{O_k}, a_k)/\tau\big).$$

  Higher τ = more uniform over actions; lower τ = more concentrated (greedy).

- As τ grows, the policy reacts less to its state input, so the policy sensitivity matrix vanishes:

$$\tau \to \infty \implies \Pi(\pi) \to 0 \implies H^\pi = E^s + E^a \Pi(\pi) \to E^s.$$

  If the environment alone is stable ($\rho(E^s) < 1$), high entropy **guarantees** locality.

- Conversely, $\tau \to 0$ makes the policy deterministic and greedy. Sensitivity to state can explode, driving $\rho(H^\pi) \geq 1$ and **breaking EDP**.

Temperature τ is a provable locality knob

- A **softmax policy** at temperature τ:

$$\pi_k(a_k \mid s_{O_k}) \;\propto\; \exp\big(g_k(s_{O_k}, a_k)/\tau\big).$$

  Higher τ = more uniform over actions; lower τ = more concentrated (greedy).

- As τ grows, the policy reacts less to its state input, so the policy sensitivity matrix vanishes:

$$\tau \to \infty \implies \Pi(\pi) \to 0 \implies H^\pi = E^s + E^a \Pi(\pi) \to E^s.$$

  If the environment alone is stable ($\rho(E^s) < 1$), high entropy **guarantees** locality.

- Conversely, $\tau \to 0$ makes the policy deterministic and greedy. Sensitivity to state can explode, driving $\rho(H^\pi) \geq 1$ and **breaking EDP**.

- In practice: start high-τ (system is local, run cheap $\kappa = 1$), then **anneal** τ downward; increasing κ as the policy tightens and coupling grows.

> τ is a **dial** on $\rho(H^\pi)$ — and therefore on locality and the required neighborhood size κ.

The locality–optimality tradeoff

High τ: calm, local

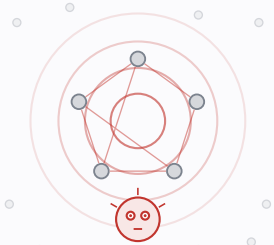


cheap κ, but exploratory

anneal

$\tau \downarrow$

Low τ: greedy, global



optimal, but coupling can explode

The locality–optimality tradeoff

High τ: calm, local

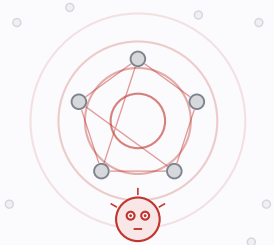


cheap κ, but exploratory

anneal

$\tau \downarrow$

Low τ: greedy, global



optimal, but coupling can explode

Start **high-τ** (local, cheap κ). Anneal toward greedy — grow κ as $\rho(H^\pi)$ rises.

Algorithm: Localized Policy Improvement (LPI)

Two phases, repeated — evaluation then improvement

Phase 1 — Evaluate

- Compute $H^\pi = E^s + E^a \Pi(\pi)$ and $\delta(r^\pi)$. Run κ **rounds of message passing**:

$$\hat{\delta}^{(\kappa)} = \sum_{t=0}^{\kappa} \big((H^\pi)^\top\big)^t \delta(r^\pi).$$

  This is the **locality certificate** directly from Theorem 3's Neumann series, truncated.

- Build the **surrogate value function**:

$$\hat{h}_\kappa^\pi = \sum_{t=0}^{\kappa} (T^\pi)^t (r^\pi - \bar{r}^\pi) - c_\kappa.$$

  Bias: $\|\hat{h}_\kappa^\pi - h^\pi\|_\infty = O(\lambda^{\kappa+1})$.

Phase 2 — Improve

Algorithm: Localized Policy Improvement (LPI)

Two phases, repeated — evaluation then improvement

Phase 1 — Evaluate

- Compute $H^\pi = E^s + E^a \Pi(\pi)$ and $\delta(r^\pi)$. Run κ **rounds of message passing**:

$$\hat{\delta}^{(\kappa)} = \sum_{t=0}^{\kappa} \big((H^\pi)^\top\big)^t \delta(r^\pi).$$

This is the **locality certificate** directly from Theorem 3's Neumann series, truncated.

- Build the **surrogate value function**:

$$\hat{h}_\kappa^\pi = \sum_{t=0}^{\kappa} (T^\pi)^t (r^\pi - \bar{r}^\pi) - c_\kappa.$$

Bias: $\|\hat{h}_\kappa^\pi - h^\pi\|_\infty = O(\lambda^{\kappa+1})$.

Phase 2 — Improve

- For each agent k in turn, compute local advantage logits from $\hat{h}_\kappa^\pi$ on its κ-hop neighbourhood.
- Take a **KL-proximal step**:

$$\pi_k^{\text{new}}(\cdot \mid s_{O_k}) \;\propto\; \pi_k(\cdot \mid s_{O_k})\, e^{\,g_k^\pi(s,\cdot)/\tau}.$$

One agent at a time; all others held fixed.

Phase 1 uses only H^π and T^π — objects already in the theory. Phase 2 is a local softmax update. Monotone progress up to bias $O(\lambda^{\kappa+1})$. (Thm 7)

Takeaways

Decomposition

$C^\pi \preceq E^s + E^a\Pi$: physics vs. logic

Spectral condition

$\rho(H^\pi) < 1$: strictly tighter; $458 \to 7$

A provable knob

temperature τ tunes locality.

Takeaways

Decomposition
$C^\pi \preceq E^s + E^a\Pi$: physics vs. logic

Spectral condition
$\rho(H^\pi) < 1$: strictly tighter; $458 \to 7$

A provable knob
temperature τ tunes locality.

Lesson?

"A calm policy can tame a chaotic world!"

Open directions

- **Beyond the oracle.** Can we remove the oracle assumption and learn from κ-hop data alone, with guarantees that survive?

Open directions

- **Beyond the oracle.** Can we remove the oracle assumption and learn from κ-hop data alone, with guarantees that survive?
- **Learning E^s, E^a, $\Pi(\pi)$ from data.** These matrices are defined by the environment and the policy but neither is known. How do you estimate them from trajectories?

Open directions

- **Beyond the oracle.** Can we remove the oracle assumption and learn from κ-hop data alone, with guarantees that survive?
- **Learning E^s, E^a, $\Pi(\pi)$ from data.** These matrices are defined by the environment and the policy but neither is known. How do you estimate them from trajectories?
- **Maintaining locality during training.** The spectral condition $\rho(H^\pi) < 1$ is for a fixed π. Can we guarantee it is preserved as the policy updates?

Open directions

- **Beyond the oracle.** Can we remove the oracle assumption and learn from κ-hop data alone, with guarantees that survive?
- **Learning E^s, E^a, $\Pi(\pi)$ from data.** These matrices are defined by the environment and the policy but neither is known. How do you estimate them from trajectories?
- **Maintaining locality during training.** The spectral condition $\rho(H^\pi) < 1$ is for a fixed π. Can we guarantee it is preserved as the policy updates?
- **Adaptive communication budget.** Can κ be scheduled automatically, i.e., growing only as the policy tightens and coupling increases?

Open directions

- **Beyond the oracle.** Can we remove the oracle assumption and learn from κ-hop data alone, with guarantees that survive?
- **Learning E^s, E^a, $\Pi(\pi)$ from data.** These matrices are defined by the environment and the policy but neither is known. How do you estimate them from trajectories?
- **Maintaining locality during training.** The spectral condition $\rho(H^\pi) < 1$ is for a fixed π. Can we guarantee it is preserved as the policy updates?
- **Adaptive communication budget.** Can κ be scheduled automatically, i.e., growing only as the policy tightens and coupling increases?
- **A phase transition?** As coupling grows, does locality break down sharply, or gradually? Is there a critical threshold?

Thank you