Distributed Q-Learning with State Tracking for Multi-agent Networked Control

Hang Wang[1], Sen Lin[1], Hamid Jafarkhani[2], Junshan Zhang[1]

[1]Arizona State University, [2]University of California, Irvine

Full paper

AAMAS 2021 LONDON, UK

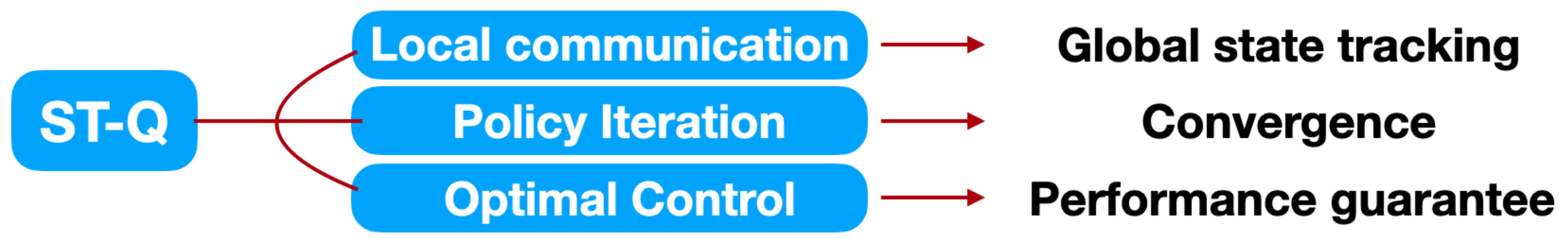
ASU UCI

Summary

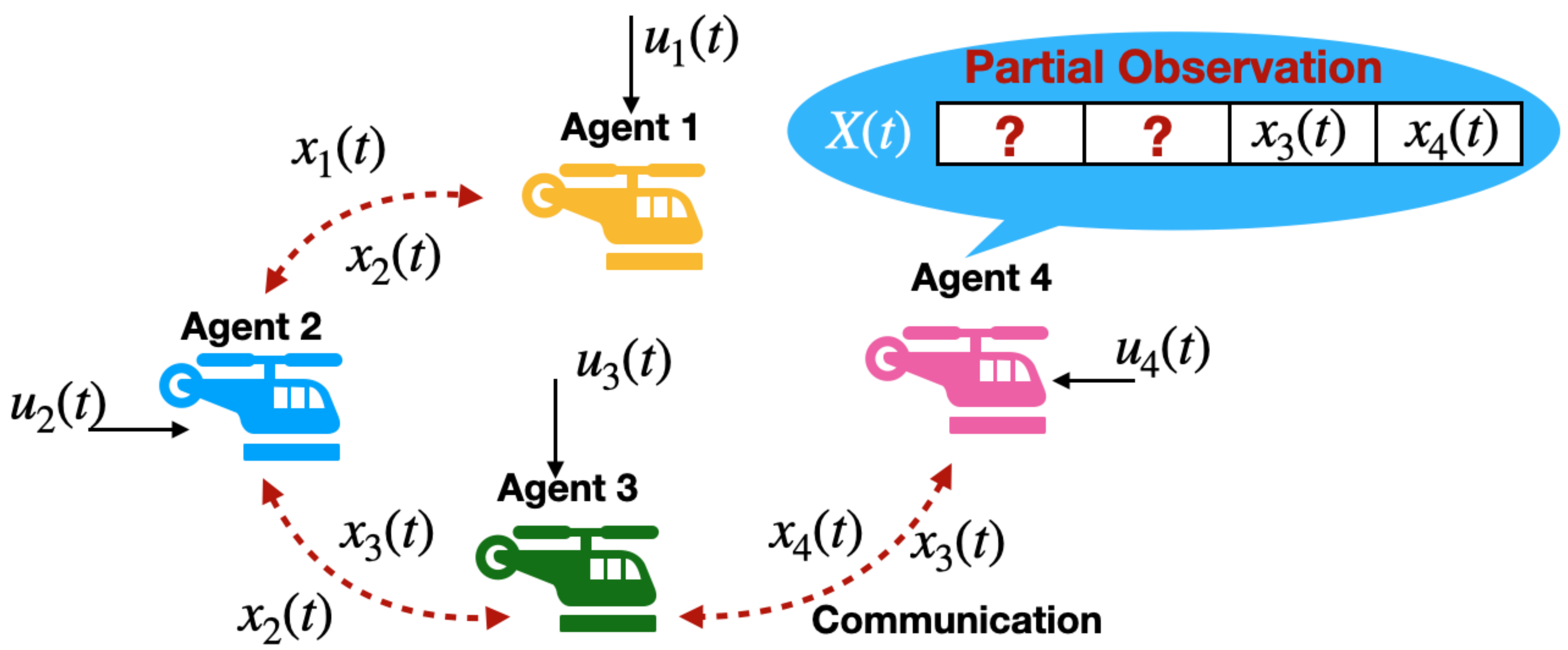
■ **Distributed Linear Quadratic Regulator (LQR) Control**

- unknown dynamics
- no central coordinator
- limited communication
- partial state observation

■ **State Tracking based Q learning (ST-Q) Algorithm**



Problem Setup



■ **Multi-agent system:** L agents

■ **Unknown LTI system:** $x_i(t+1) = \sum_{j=1}^{L} A_{ij} x_j(t) + B_i u_i(t)$

■ **Quadratic cost:** $g_i(t) = x_i(t)^\top P_i x_i(t) + u_i(t)^\top R_i u_i(t)$

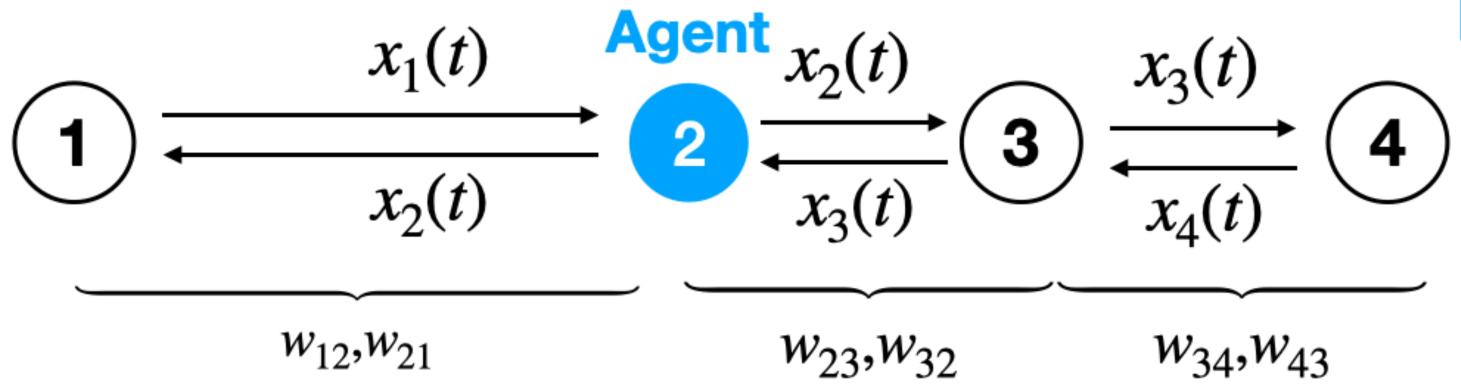
■ **Communication without a central coordinator**

■ **Linear feedback controller:** $K_i \ (u_i(t) = K_i X(t))$

■ **Goal:** Find controllers for each agent that minimizes infinite-horizon cost of the whole system:

$$\min_{K_1,\cdots,K_L} \sum_{i=1}^{L} \sum_{\tau_0}^{\infty} g_i(\tau)$$
$$\text{s.t. } X(t) \text{ is partially observed}$$

State Tracking-Q Learning



$$Z_i(t) = [\bar{x}_{i1}(t) \quad \bar{x}_{i2}(t) \quad \bar{x}_{i3}(t) \quad \bar{x}_{i4}],\ i \in [L]$$

$$Z_2(t) = [x_{21}(t) \quad x_{22}(t) \quad x_{23}(t) \quad ?]$$

■ **Step 1: Exchange current state with one-hop neighbor(s), e.g.,**

$$x_1(t),\ x_3(t) \longrightarrow \text{Agent 2}$$

■ **Step 2: Weight the non-neighbor state estimation, e.g.,**

$$Z_1(t),\ Z_3(t) \longrightarrow \text{Agent 2}$$

Estimation towards Agent 4:

$$\bar{x}_{24}(t) = w_{21}\bar{x}_{14}(t) + w_{23}\bar{x}_{34}(t)$$

Estimation from Agent 1, Agent 3

▶ **Q-factor (Bellman Equation):**

$$Q_i(x_i(t), K_i X(t)) = g_i(t) + Q_i(x_i(t+1), K_i X(t+1))$$

▶ **Linear structure of the Q:**

$$Q_i(x_i(t), u_i(t)) = [X(t); u_i(t)]^\top H_i [X(t); u_i(t)] = y_i(t)^\top \theta_i,$$
$$y_i(t) = [x_1^2(t), x_1(t)x_2(t), \cdots, x_L(t)u_i(t), u_i^2(t)]$$

▶ **Linear regression problem:**

$$g_i(x_i(t), u_i(t)) = (y_i(t) - y_i(t+1))^\top \theta_i \triangleq \phi_i(t)^\top \theta_i$$

Repeat $q = 1, \cdots$ **(Policy Iteration)**
  For agent $i = 1, \cdots, L$
    For $p = 1, \cdots, N$ **(Policy Evaluation)**
      Apply current policy: $u_i(t) = -K_{iq} Z_i(t) + \text{noise}$
      Measure $x_i(t+1)$
      State Tracking $Z_i(t+1)$ **(State Tracking)**
      Update parameter estimation $\theta_{iq}(p)$ **(SGD/RLS)**
    End for
  End for
  For agent $i = 1, \cdots, L$
    Obtain H_{iq} from $\theta_{iq}(p)$ **(Policy Improvement)**
    Update $K_{i(q+1)} = -H_{iq,22}^{-1} H_{iq,21}$
End for

Performance

Assumption

▶ **System parameters are stabilizable**

▶ **Communication graph is connected**

▶ **Weight matrix is doubly stochastic**

▶ **Excitation noise is decaying**

Convergence

■ $Z_i(t) \rightarrow X(t)$ ■ $K_i \rightarrow K_i^*$