

Payoff Mechanism Design for Coordination in Multi-Agent Task Allocation Games

Shinkyu Park and Julian Barreiro-Gomez

Abstract—We investigate a multi-agent decision-making problem where a large population of agents is responsible for carrying out a set of assigned tasks. The amount of jobs in each task varies over time governed by a dynamical system model. Each agent needs to select one of the available strategies to take on one or more tasks. Since each strategy allows an agent to perform multiple tasks at a time, possibly at distinct rates, the strategy selection of the agents needs to be coordinated. We formulate the problem using the population game formalism and refer to it as the *task allocation game*. We discuss the design of a decision-making model that incentivizes the agents to coordinate in the strategy selection process.

As key contributions, we propose a method to find a payoff-driven decision-making model, and discuss how the model allows the strategy selection of the agents to be responsive to the amount of remaining jobs in each task while asymptotically attaining the optimal strategies. Leveraging analytical tools from feedback control theory, we derive technical conditions that the model needs to satisfy, which are used to construct a numerical approach to compute the model. We validate our solution through simulations to highlight how the proposed approach coordinates the agents in task allocation games.

I. INTRODUCTION

We investigate task allocation games to study coordination in repeated strategic interactions in a large population of agents. Consider that there is a finite number of tasks to be carried out by the agents. We quantify the amount of jobs remaining in each task with a positive variable, and every agent can select one of the available strategies at a time to take on one or more tasks. The main objective is to design a decentralized decision-making model that allows the agents to coordinate and minimize remaining jobs in all tasks.

Task allocation games are relevant to engineering applications. For instance, in multi-robot resource retrieval [1], [2], a team of multiple robots is tasked with searching and collecting target resources across partitioned areas in a given environment. Each task can be defined as collecting resources from an area and the strategy selection refers to taking one of the tasks. In target tracking applications [3], a group of mobile units with heterogeneous sensing capabilities are deployed to collect data about the states of multiple targets of our interests. Based on the type of equipped sensors, each

mobile unit can collect different sets of data on the targets' states. A task is defined as collecting data on a portion of the target states and a strategy specifies which pair of available sensors a mobile unit can equip. In both scenarios, the amount of resources to collect and the data to gather vary depending on past strategy selection of the agents and also on environmental changes and target dynamics.

To design a model for the agent strategy selection in such engineering applications, we investigate task allocation in dynamically changing environments. Multi-agent task allocation problems have been widely studied across various research communities [4]–[10]. A game-theoretic approach to the problem using replicator dynamics is investigated in [8]. The authors of [5], [6] use the hedonic game to study the coordination of multiple agents in task allocation. Applications of population game approaches to address task allocation in swarm robotics [9] and the control of a water distribution system [10] are discussed. Also, relevant to the task allocation game that we investigate in this work, whose formalism is defined in a state space, the state-based potential game has been studied in [11], and the design of state-based game to solve distributed optimization is proposed in [12].

A majority of existing works assume that the environment underlying the game is static and aim to find the optimal task allocation. In contrast, we study the design of a decision-making model under which the agents can repeatedly switch among multiple tasks to minimize remaining jobs in the tasks. We adopt the population game formalism [13] to state the problem and to study the decision-making model design. The model prescribes how the agents take on a given set of tasks and how the agents should switch among the tasks by revising their strategy selection to asymptotically attain optimality. We consider that each agent in the population is given a set of n strategies to carry out assigned tasks where we denote the agents' strategy profile – the distribution of the agents' strategy selection – by a non-negative vector $x = (x_1, \dots, x_n)$. Remaining jobs associated with m tasks are denoted by a non-negative vector $q = (q_1, \dots, q_m)$ for which a dynamic model describes how q changes – both growth by environmental changes and reduction by the agents – based on the agents' strategy selection x .

Based on the evolutionary dynamics framework [13], we specify a decentralized decision-making model that allows individual agents to revise their strategy selection based on a payoff vector $p = (p_1, \dots, p_n)$, where each p_i is the payoff an agent receives when it selects the i -th strategy. As the main contribution, we design a payoff mechanism to define how p should depend on q to encourage the

Park's work was supported by funding from King Abdullah University of Science and Technology (KAUST). Barreiro-Gomez's work was supported by the Center on Stability, Instability, and Turbulence (SITE) and Tamkeen under the NYU Abu Dhabi Research Institute grant CG002.

Park is with the Electrical and Computer Engineering, King Abdullah University of Science and Technology (KAUST), Thuwal 23955, Saudi Arabia. shinkyu.park@kaust.edu.sa

Barreiro-Gomez is with NYUAD Research Institute, New York University Abu Dhabi, PO Box 129188, Abu Dhabi, United Arab Emirates. jbarreiro@nyu.edu

agents to select the tasks with more jobs to perform and asymptotically attain the minimum of a given cost $c(q)$. Applying convergence analysis tools [14]–[19] that are based on passivity theory in the population games literature, we establish conditions under which the agents’ strategy profile converges and asymptotically attain the optimal profile. We use the conditions to compute the payoff mechanism.

The paper is organized as follows. In Section II, we explain the task allocation game formulation and the main problem we address in this paper. In Section III, we present the main result on the payoff mechanism design and analysis on convergence of the agent strategy revision process to the optimal strategy profile. In Section IV, we present simulation results to illustrate our main contribution. We conclude the paper with a summary and future plans in Section V.

II. PROBLEM DESCRIPTION

Consider a large population of agents that are assigned with m tasks and are given n strategies to carry out the tasks.¹ We associate each task $j \in \{1, \dots, m\}$ with a variable $q_j \geq 0$ which quantifies the amount of jobs remaining in the task. Let $x_i \geq 0$ denote the portion of the agents selecting strategy $i \in \{1, \dots, n\}$ and a fixed positive number M to be the mass of the agent population satisfying $M = \sum_{i=1}^n x_i$.² Each agent selects one of the strategies at a time based on payoff vector $p = (p_1, \dots, p_n)$.

Let \mathbb{R}_+^n be the set of all n -dimensional vectors with non-negative entries, and let \mathbb{X}_M be the space of all feasible states $x = (x_1, \dots, x_n)$ of the population defined as $\mathbb{X}_M = \{x \in \mathbb{R}_+^n \mid \sum_{i=1}^n x_i = M\}$. Given a matrix $G \in \mathbb{R}^{n \times m}$, we represent G using its column and row vectors as follows:

$$G = \begin{pmatrix} G_1^{\text{col}} & \dots & G_m^{\text{col}} \end{pmatrix} = \begin{pmatrix} G_1^{\text{row}} \\ \vdots \\ G_n^{\text{row}} \end{pmatrix}. \quad (1)$$

A. Task Allocation Games

To investigate the task allocation problem, we formalize the problem as a large population game in which the agents select strategies to perform jobs in the assigned tasks quantified by $q = (q_1, \dots, q_m)$. The vector q varies over time based on the agents’ strategy selection and changes in the environment. Hence, each agent needs to evaluate and adaptively select a strategy based on q .

Given $x(t)$ and $q(t)$, at each time t , the following ordinary differential equation describes the rate of change of $q(t)$.

$$\dot{q}(t) = - \underbrace{\mathcal{F}(q(t), x(t))}_{\text{reduction rate}} + \underbrace{w}_{\text{growth rate}}, \quad q(0) = q_0 \in \mathbb{R}_+^m, \quad (2)$$

where $\mathcal{F} : \mathbb{R}_+^m \times \mathbb{R}_+^n \rightarrow \mathbb{R}_+^m$ is a continuously differentiable mapping³ that defines the *reduction rate*, which quantifies how fast the agents adopting strategy profile x reduce q , and

¹The number of tasks is not necessarily the same as that of available strategies, i.e., $m \neq n$.

²Considering the population state x as control input to (2), the population mass M can be interpreted as a limit on the control input.

³To have the reduction rate mapping defined for any population mass M , we define the domain of \mathcal{F} as $\mathbb{R}_+^m \times \mathbb{R}_+^n$.

the constant vector $w = (w_1, \dots, w_m) \in \mathbb{R}_+^m$ represents the *growth rate* for q due to environmental changes. To ensure that the positive orthant \mathbb{R}_+^m is forward-invariant for $q(t)$ under (2), each \mathcal{F}_i of $\mathcal{F} = (\mathcal{F}_1, \dots, \mathcal{F}_m)$ satisfies $\mathcal{F}_i(q, x) \leq w_i$ if $q_i = 0$. For notational convenience, let us define \mathbb{O} as the set of stationary points of (2), i.e.,

$$\mathbb{O} = \{(q, x) \in \mathbb{R}_+^m \times \mathbb{X}_M \mid \mathcal{F}(q, x) = w\}. \quad (3)$$

We make the following assumption on the mapping \mathcal{F} .

Assumption 1: The reduction rate \mathcal{F}_i for each task i depends only on its associated variable $q_i(t)$ and the agent strategy selection $x(t)$, and increases as does $q_i(t)$. For instance, in the resource retrieval application discussed in Section I, when there is a larger volume of resources spread out across the areas, the robots would need to travel a shorter distance on average to locate and retrieve the resources and hence given a fixed strategy profile x , the variable $q_i(t)$ decreases at a faster rate. We formalize such assumptions as $\frac{\partial \mathcal{F}_i}{\partial q_i}(q, x) = 0$ if $i \neq j$ and $\frac{\partial \mathcal{F}_i}{\partial q_i}(q, x) > 0$. \square

According to Assumption 1, we represent the reduction rate as $\mathcal{F}(q, x) = (\mathcal{F}_1(q_1, x), \dots, \mathcal{F}_m(q_m, x))$, where for fixed x , each \mathcal{F}_i is an increasing function of q_i .

Remark 1: Suppose that given x in \mathbb{X}_M , there is q in \mathbb{R}_+^m satisfying $\mathcal{F}(q, x) = w$. By Assumption 1, q is unique. \square

The following examples illustrate how the dynamic game model (2) can be adopted in control systems applications.

Example 1 (Multi-Robot Resource Collection [1]): Let $m = n$ and $\mathcal{F} = (\mathcal{F}_1, \dots, \mathcal{F}_n)$ be defined as

$$\mathcal{F}_i(q_i, x_i) = R_i \frac{\exp(\alpha_i q_i) - 1}{\exp(\alpha_i q_i) + 1} x_i^{\beta_i}, \quad (4)$$

where R_i , α_i , and β_i are positive constants. The parameter R_i represents the maximum reduction rate associated with strategy i , and α_i and β_i are coefficients specifying how the reduction rate \mathcal{F}_i depends on q_i and x_i , respectively. Note that each function \mathcal{F}_i satisfies $\mathcal{F}_i(0, x) = 0$ and Assumption 1. Here, $m = n$ and only the agent selecting strategy i can reduce q_i associated with task i .

Example 2 (Heterogeneous Sensor Scheduling [3]): We adopt the model (2) as an abstract description of how mobile units’ sensor scheduling affects the uncertainty reduction in estimating states of multiple targets. Let $m < n$ and $\mathcal{F} = (\mathcal{F}_1, \dots, \mathcal{F}_m)$ be defined as

$$\mathcal{F}_i(q_i, x) = \sum_{j \in \mathbb{N}_i} R_i \frac{\exp(\alpha_i q_i) - 1}{\exp(\alpha_i q_i) + 1} x_j^{\beta_i}, \quad (5)$$

where \mathbb{N}_i denotes the set of the strategies (available sensor configurations of a mobile unit) that can collect data on the state of the i -th target. The parameters R_i , α_i , β_i have the same interpretation as in Example 1. Unlike the previous example, the strategies are defined to allow the agents to reduce multiple task-associated variables of q .

B. Agent Strategy Revision Model

Our model is based on the evolutionary dynamics framework [13] in which the *strategy revision protocol* $\varrho_i^\theta : \mathbb{R}_+^n \rightarrow \mathbb{R}_+$ determines an agent’s strategy revision based on the

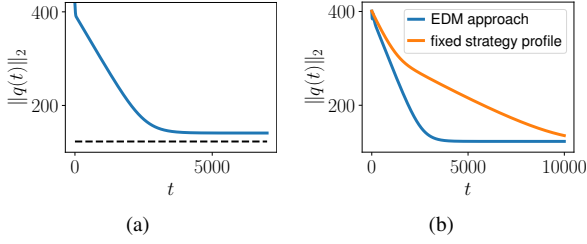


Fig. 1. Graphs depicting trajectories $\|q(t)\|_2$, $t \geq 0$ determined by (2) and (7) using Example 1. The parameters of (2) are defined as $m = n = 4$, $M = 1$, $R_i = 3.5$, $\alpha_i = 0.05$, $\beta_i = 1$, $w = (0.05, 0.25, 1.00, 2.00)$, and those of (6) as $\theta = x^*$, $\eta = 0.001$, where $(q^*, x^*) \in \mathbb{O}$ is the equilibrium state minimizing $\max_{1 \leq i \leq 4} q_i$. In (a), the dotted black line is the minimum 2-norm achievable when the payoff mechanism $p = Gq$ is optimally designed, and the blue line represents the trajectory $\|q(t)\|_2$, $t \geq 0$ when the population state is determined by (7) with $p = q$. In (b), the blue and orange lines represent the trajectories $\|q(t)\|_2$, $t \geq 0$ when the population state is determined by (7) and is fixed to x^* , respectively.

payoff vector $p \in \mathbb{R}^n$, where $\theta = (\theta_1, \dots, \theta_n) \in \mathbb{X}_M$ is a parameter of the protocol. We adopt the Kullback-Leibler Divergence Regularized Learning (KLD-RL) protocol [20], [21] to define $\varrho_i^\theta(p)$ as

$$\varrho_i^\theta(p) = \frac{\theta_i \exp(\eta^{-1} p_i)}{\sum_{l=1}^n \theta_l \exp(\eta^{-1} p_l)}, \quad (6)$$

where $\eta > 0$. The protocol $\varrho_i^\theta(p)$ describes the probability of an agent switching to strategy i given p and θ . Note that the smaller the value of η , the more the strategy revision depends on the value of p .

Each agent is given an opportunity to revise its strategy selection at each jump time of an independent and identically distributed Poisson process, and uses the protocol to select a new strategy or keep its current strategy selection. Since the strategy revision of individual agents only depends on the payoff vector and takes place independently of each other, their decision-making is decentralized and the coordination among them occurs implicitly through their decision-making model. Based on discussions in [13, Chapter 4], as the number of agents in the population tends to infinity, the following ordinary differential equation describes how each component of $x(t) = (x_1(t), \dots, x_n(t))$ evolves over time.

$$\begin{aligned} \dot{x}_i(t) &= \mathcal{V}_i^\theta(p(t), x(t)) \\ &= \sum_{j=1}^n x_j(t) \varrho_j^\theta(p(t)) - x_i(t) \sum_{j=1}^n \varrho_j^\theta(p(t)). \end{aligned} \quad (7)$$

We refer to (7) as the Evolutionary Dynamics Model (EDM).

Note that at an equilibrium state (p^*, x^*) of the EDM (7) under the KLD-RL protocol (6), if $\theta = x^*$, the following implication holds:

$$x_i^* > 0 \implies p_i^* = \max_{1 \leq j \leq n} p_j^*. \quad (8)$$

Eq. (8) means that every agent receives the highest payoff at (p^*, x^*) if the parameter θ of (6) is the same as x^* .

Given the protocol ϱ_i^θ as in (6), we aim to design a payoff mechanism for the agents to asymptotically adopt the optimal strategy profile that minimizes a given cost $c(q)$. For instance, in Example 1, if we design the payoff mechanism as $p = q$, the robots would select strategy i to take on task i and asymptotically minimize $\lim_{t \rightarrow \infty} \max_{1 \leq i \leq m} q_i(t)$, as

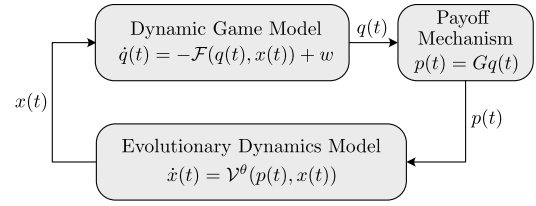


Fig. 2. Feedback model in the task allocation game.

discussed in [1]. However, in many applications, such one-to-one correspondence between tasks and available strategies may not exist, and depending on the cost we want to minimize, such a simple payoff mechanism would not be the best design choice as we illustrate in Figure 1(a).

In addition, since the payoff mechanism depends on the vector $q(t)$, the mechanism would incentivize the agents to take on the tasks with larger $q_i(t)$. Hence, compared to other models that directly control the population state $x(t)$ to the optimal state x^* (for instance, the model proposed in [9]), our strategy revision model is more responsive to changes of $q(t)$ and hence reduces the task-associated variables $q(t)$ at a faster rate as we depict in Figure 1(b).

Two examples of the cost function we consider are

- (square of) the 2-norm of q : $c(q) = \sum_{i=1}^m q_i^2$, and
- the ∞ -norm of q : $c(q) = \max_{1 \leq i \leq m} q_i$.

For the payoff mechanism design, we consider a linear model defined by a matrix $G \in \mathbb{R}^{n \times m}$ as follows:

$$p = Gq. \quad (9)$$

Our main problem investigates finding the matrix G that allows the agents to asymptotically minimize the cost $c(q(t))$. We formally state the problem as follows.

Problem 1: Given the dynamic model (2) of the task allocation game and the EDM (7), compute the payoff matrix G under which the cost $c(q(t))$ is asymptotically minimized.

III. PAYOFF MATRIX DESIGN

By interconnecting the dynamic model of the game (2), the payoff mechanism (9), and the EDM (7) with (6) as its revision protocol, as illustrated in Figure 2, we can write the state equation of the resulting closed-loop model as follows:

$$\begin{cases} \dot{q}(t) &= -\mathcal{F}(q(t), x(t)) + w \\ \dot{p}(t) &= Gq(t) \end{cases} \quad (10a)$$

$$\dot{x}_i(t) = M \frac{\theta_i \exp(\eta^{-1} p_i(t))}{\sum_{l=1}^n \theta_l \exp(\eta^{-1} p_l(t))} - x_i(t). \quad (10b)$$

Given an initial condition $(q(0), x(0)) \in \mathbb{R}_+^m \times \mathbb{X}_M$, we assume the closed-loop model (10) has a unique solution. Let \mathbb{S} be the set of equilibrium states of (10). The proper design of G should ensure that the following two conditions hold.

- (R1) The state $(q(t), x(t))$ converges to the stationary points of (10a), i.e., it holds that $\lim_{t \rightarrow \infty} \inf_{(r,z) \in \mathbb{O}} (\|q(t) - r\|_2 + \|x(t) - z\|_2) = 0$.
- (R2) When the closed-loop model (10) reaches an equilibrium state (q^*, x^*) , it attains the minimum cost, i.e., $c(q^*) = \inf_{(q,x) \in \mathbb{O}} c(q)$.

We adopt passivity tools [15], [17] to find technical conditions under which (R1) and (R2) are attained and use the conditions to design the payoff matrix G . The critical step in the convergence analysis (R1) is in establishing passivity for both (10a) and (10b) by finding a so-called δ -storage function for (10a) and δ -antistorage function for (10b).⁴ Then, by constructing a Lyapunov function using the two storage functions, we establish convergence results for (10).

To proceed, by [21, Lemma 3], (10b) is δ -passive and has the δ -storage function $\mathcal{S}^\theta : \mathbb{R}^n \times \mathbb{X}_M \rightarrow \mathbb{R}_+$ given by

$$\mathcal{S}^\theta(p, x) = \max_{z \in \mathbb{X}_M} (p^T z - \eta \mathcal{D}(z \| \theta)) - (p^T x - \eta \mathcal{D}(x \| \theta)), \quad (11)$$

where $\mathcal{D}(\cdot \| \cdot)$ is the KL divergence. Note that \mathcal{S}^θ satisfies

$$\mathcal{S}^\theta(p, x) = 0 \Leftrightarrow \mathcal{V}^\theta(p, x) = 0 \Leftrightarrow \nabla_x^T \mathcal{S}^\theta(p, x) \mathcal{V}^\theta(p, x) = 0 \quad (12a)$$

$$\begin{aligned} \mathcal{S}^\theta(p(t), x(t)) - \mathcal{S}^\theta(p(t_0), x(t_0)) \\ \leq \int_{t_0}^t \dot{p}^T(\tau) \dot{x}(\tau) d\tau, \quad \forall t \geq t_0 \geq 0 \end{aligned} \quad (12b)$$

for any payoff vector trajectory $p(t)$, $t \geq 0$. The mapping $\mathcal{V}^\theta = (\mathcal{V}_1^\theta, \dots, \mathcal{V}_n^\theta)$ is the vector field of the EDM (7).

The dynamic game model (2) is qualified as δ -antipassive [17] if there is a δ -antistorage function $\mathcal{L} : \mathbb{R}_+^m \times \mathbb{X}_M \rightarrow \mathbb{R}_+$ satisfying the following two conditions:

$$\begin{aligned} \mathcal{L}(q, x) = 0 &\Leftrightarrow \mathcal{F}(q, x) = w \\ &\Leftrightarrow \nabla_q^T \mathcal{L}(q, x) (\mathcal{F}(q, x) - w) = 0 \end{aligned} \quad (13a)$$

$$\begin{aligned} \mathcal{L}(q(t), x(t)) - \mathcal{L}(q(t_0), x(t_0)) \\ \leq - \int_{t_0}^t \dot{q}^T(\tau) G^T \dot{x}(\tau) d\tau, \quad \forall t \geq t_0 \geq 0, \end{aligned} \quad (13b)$$

where (13b) needs to hold for any given population state trajectory $x(t)$, $t \geq 0$. According to (13a), the function $\mathcal{L}(q, x)$ can be used to measure how far the state (q, x) is from the equilibrium of (10a). By their respective definitions [17], both \mathcal{S}^θ and \mathcal{L} need to be continuously differentiable.

Recall \mathbb{O} given as in (3). For $(q^*, x^*) \in \mathbb{O}$ satisfying

$$x_i^* > 0 \implies p_i^* = \max_{1 \leq j \leq n} p_j^*, \quad \forall i \in \{1, \dots, n\} \quad (14)$$

with $p^* = Gq^*$, let us assign $\theta = x^*$ for (10b). We can establish the following lemma.

Lemma 1: If the dynamic game model (10a) is δ -antipassive, then given that $q(t)$, $t \geq 0$ is bounded, the state $(q(t), x(t))$ of the closed-loop model (10) converges to \mathbb{S} . Also (q^*, x^*) is the equilibrium state of (10) for all $\eta > 0$.

The proof of the lemma is given in Appendix of [22]. Resorting to Lemma 1, to meet the requirements (R1) and (R2), we need to construct the payoff matrix G such a way that (10a) becomes δ -antipassive and $(q^*, x^*) \in \mathbb{O}$ minimizing $c(q)$ is an equilibrium state of (10). The following theorem states the technical conditions on G that ensure (R1)

⁴We refer to [17, Definition 10] and [17, Definition 12], respectively, for the formal definitions of passivity for (10a) and (10b).

and (R2). To state the theorem, we define a continuously differentiable mapping $g : \mathbb{R}_+^m \rightarrow \mathbb{R}_+^n$ that maps any $q \in \mathbb{R}_+^m$ to $y = g(q)$ satisfying $\mathcal{F}(q, y) = w$.⁵ The statement of the theorem holds if such g exists.

Theorem 1: Let us define

$$h_i(q, x) = (\mathcal{F}_i(q_i, x) - w_i) (x - g(q)), \quad i \in \{1, \dots, m\}$$

and let (q^*, x^*) be the stationary point of (10a) attaining the minimum cost $\inf_{(q, x) \in \mathbb{O}} c(q)$. Suppose the matrix G satisfies

$$G \nabla_x \mathcal{F}(q, x) = \nabla_x^T \mathcal{F}(q, x) G^T, \quad \forall (q, x) \in \mathbb{R}_+^m \times \mathbb{R}_+^n \quad (15a)$$

$$h_i^T(q, x) G_i^{\text{col}} > 0, \quad \forall (q, x) \notin \mathbb{O}, \quad \forall i \in \{1, \dots, m\} \quad (15b)$$

$$(G_i^{\text{row}} - G_j^{\text{row}}) x_i^* q_j^* \geq 0, \quad \forall i, j \in \{1, \dots, n\}, \quad (15c)$$

where G_i^{col} and G_i^{row} are the column and row vectors of G defined as in (1), respectively. The dynamic game model (10a) is δ -antipassive and (q^*, x^*) is an equilibrium state of (10) with $\theta = x^*$ for any $\eta > 0$.

The proof of the theorem is given in Appendix of [22]. Under the condition (15b), whenever $q_i(t)$ is increasing, i.e., $\dot{q}_i(t) = -\mathcal{F}_i(q_i(t), x(t)) + w_i > 0$, the matrix G incentivizes the agents to revise their strategies toward $g(q)$, which is the strategy profile required to make the rate $\dot{q}(t)$ to zero. In other words, G is designed to encourage the agents to select strategies that reduce the rate $\dot{q}(t)$.

Proposition 1: Let (q^*, x^*) be the stationary point of (10a) attaining the minimum cost $\inf_{(q, x) \in \mathbb{O}} c(q)$. Consider the closed-loop model (10) for which $\theta = x^*$ and the payoff matrix G satisfies (15). As the parameter η of (10b) increases, (q^*, x^*) becomes the unique equilibrium state of (10). In other words, it holds that $\lim_{\eta \rightarrow \infty} \sup_{(\bar{q}, \bar{x}) \in \mathbb{S}} \mathcal{D}(\bar{x} \| x^*) = 0$, where \mathbb{S} is the set of equilibrium states of (10).

The proof of the proposition is provided in Appendix of [22]. In conjunction with Lemma 1 and Theorem 1, Proposition 1 implies that as η becomes sufficiently large, the state trajectory $(q(t), x(t))$, $t \geq 0$ converges to near the optimal state (q^*, x^*) . According to (6), we note that smaller η is desired to make the agent strategy revision responsive to changes in $p(t)$ and also in $q(t)$. Hence, a good practice is to use smaller η at the beginning of the task allocation game, and if needed, as $\dot{q}(t)$ goes to zero, the agents can gradually increase the value of η to ensure that $x(t)$ converges to x^* .

IV. SIMULATIONS

We use Examples 1 and 2 to illustrate our main results and discuss how the cost function and parameters of the dynamic model (2) affect the payoff matrix design. In both examples, we select the following fixed parameters $M = 1$, $R_i = 3.5$, $\alpha_i = 0.05$, and $\beta_i = 1$ for (4) and (5), and $\eta = 0.001$ for (10b).⁶ We use two different cost functions $c(q)$ for Example 1 and two distinct growth rates w for Example 2.

⁵We remark that $g(q)$ does not necessarily belong to \mathbb{X}_M . We interpret $g(q)$ as the strategy profile that attains the equilibrium state for a given q when there is no limit on the population mass M .

⁶We select $\eta = 0.001$ as all population state trajectories in the simulations converge to the optimal x^* with the small positive η .

A. Computation of G

We explain the steps to compute G . First, note that (15a) is satisfied if G has the following structures:

- 1) For Example 1, $G_{ij} = 0$ if $i \neq j$.
- 2) For Example 2, $G_{ij} = 0$ if $i \notin \mathbb{N}_j$ and $G_{ij} = G_j$ otherwise, where G_j is a real number.

Then, we find $(q^*, x^*) \in \mathbb{O}$ that minimizes the cost function $c(q)$ using the following optimization.

$$\min_{(q,x) \in \mathbb{R}_+^m \times \mathbb{X}_M} c(q) \text{ subject to } \mathcal{F}(q, x) = w. \quad (16)$$

Note that since \mathcal{F} is a nonlinear mapping, the optimization can be non-convex and the solution we find is locally optimal.

Once we find (q^*, x^*) , we compute the matrix G satisfying (15) for which we first need to find the mapping g . Instead of explicitly finding g , we draw random samples $\{(q_s, x_s)\}_{s=1}^S \subset \mathbb{R}_+^m \times \mathbb{X}_M$ and find $y_s \in \mathbb{R}_+^n$ that minimizes $\|\mathcal{F}(q_s, y_s) - w\|_2^2$ for each sample (q_s, x_s) . Note that assuming $\nabla_x \mathcal{F}(q, x)$ has full rank at (q_s, y_s) , which is the case in both examples, the minimizer y_s satisfies $\mathcal{F}(q_s, y_s) = w$.

As the last step, the design of G can be formulated as the following linear programming:

$$\begin{aligned} \min_{G \in \mathbb{R}^{n \times m}} & 1 \\ \text{subject to} & (\mathcal{F}_i(q_{s,i}, x_s) - w_i) (x_s - y_s)^T G_i^{\text{col}} > 0, \\ & \forall i \in \{1, \dots, m\}, \forall s \in \{1, \dots, S\} \\ & (G_i^{\text{row}} - G_j^{\text{row}}) x_i^* q_j^* \geq 0, \forall i, j \in \{1, \dots, n\}, \end{aligned} \quad (17)$$

where $q_{s,i}$ is the i -th element of $q_s = (q_{s,1}, \dots, q_{s,m})$. Since we evaluate the condition (15b) using a finite number of sampled points $\{(q_s, x_s)\}_{s=1}^S$, we would obtain an approximate solution satisfying (15) only at the sampled points. However, as the sample size S tends to infinity, the solution G is more likely to satisfy (15) over the entire state space $\mathbb{R}_+^m \times \mathbb{X}_M$.

B. Simulation results for Example 1 ($m = 4, n = 4$)

Using the methods explained in Section IV-A, we compute the optimal state (q^*, x^*) minimizing **i)** $c(q) = \sum_{i=1}^m q_i^2$ and **ii)** $c(q) = \max_{1 \leq i \leq m} q_i$, where we use the fixed growth rate $w = (0.05, 0.25, 1.00, 2.00)$ for both cases. Then, we design the payoff matrix G using (17) as follows.

- i) For $c(q) = \sum_{i=1}^m q_i^2$,

$$G = \begin{pmatrix} 1.00 & 0.00 & 0.00 & 0.00 \\ 0.00 & 0.66 & 0.00 & 0.00 \\ 0.00 & 0.00 & 0.48 & 0.00 \\ 0.00 & 0.00 & 0.00 & 0.40 \end{pmatrix}.$$

- ii) For $c(q) = \max_{1 \leq i \leq m} q_i$,

$$G = \begin{pmatrix} 1.00 & 0.00 & 0.00 & 0.00 \\ 0.00 & 1.00 & 0.00 & 0.00 \\ 0.00 & 0.00 & 1.00 & 0.00 \\ 0.00 & 0.00 & 0.00 & 1.00 \end{pmatrix}.$$

Note that when we use the ∞ -norm to define the cost $c(q)$, the optimal design of G equally incentivizes the agents proportional to the remaining jobs q . On the other hand, when the 2-norm is used, given that the values of $q_1(t), \dots, q_4(t)$

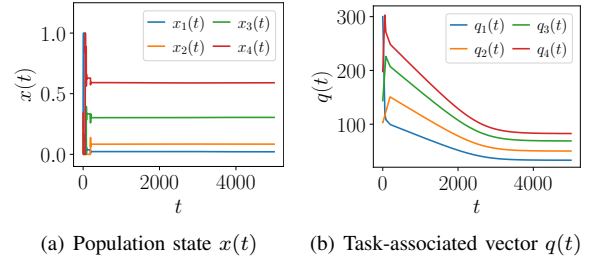


Fig. 3. Graphs depicting the trajectories of the (a) population state $x(t)$ and (b) task-associated vector $q(t)$ derived by the closed-loop model (10) in Example 1 using the cost function $c(q) = \sum_{i=1}^4 q_i^2$.

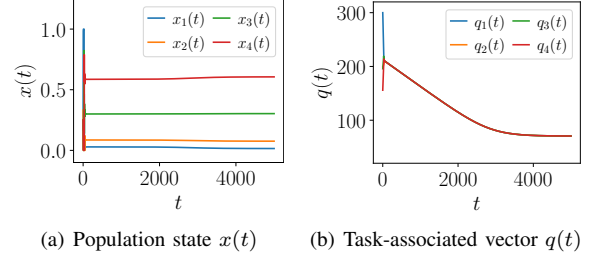


Fig. 4. Graphs depicting the trajectories of the (a) population state $x(t)$ and (b) task-associated vector $q(t)$ derived by the closed-loop model (10) in Example 1 using the cost function $c(q) = \max_{1 \leq i \leq 4} q_i$.

are equal, the payoff matrix G assigns the highest payoff to strategy 1 and the lowest payoff to strategy 4. Recall that under the pre-selected growth rate w , task 1 has the lowest growth rate and task 4 has the highest, and hence maintaining lower $q_1(t)$ is easier – it needs a less number of agents – than $q_4(t)$. Hence, under the 2-norm cost function, the agents prioritize to carry out the tasks with lower growth rates.

Figures 3 and 4 depict the resulting trajectories for $x(t)$ and $q(t)$. Notice that the population states at the equilibrium in the two cases are similar; however, the trajectories for $q(t)$ are different and, hence, so do the costs evaluated along the trajectories as we discussed in Figure 1(a). We observe that there is a large variation in the agent strategy revision at the beginning of the simulations as the agents repeatedly switch among the strategies to reduce $q_i(t)$ with a larger value.

C. Simulation results for Example 2 ($m = 4, n = 6$)

We consider that there are 4 target states and 4 types of sensors each of which can measure a single state of the target. Each mobile unit can be equipped with two types of sensors and we define a strategy based on a pair of sensors employed on a mobile unit. According to the strategy definition, we can define the set \mathbb{N}_i in (5) as the strategies that can measure the i -th target state: $\mathbb{N}_1 = \{1, 2, 3\}$, $\mathbb{N}_2 = \{1, 4, 5\}$, $\mathbb{N}_3 = \{2, 4, 6\}$, and $\mathbb{N}_4 = \{3, 5, 6\}$. We use the square of the 2-norm to define the cost function, i.e., $c(q) = \sum_{i=1}^m q_i^2$.

We design G with two distinct growth rates as follows.

- i) For $w = (0.5, 1.0, 1.5, 2.0)$,

$$G = \begin{pmatrix} 1.00 & 0.81 & 0.00 & 0.00 \\ 1.00 & 0.00 & 0.72 & 0.00 \\ 1.00 & 0.00 & 0.00 & 0.67 \\ 0.00 & 0.81 & 0.72 & 0.00 \\ 0.00 & 0.81 & 0.00 & 0.67 \\ 0.00 & 0.00 & 0.72 & 0.67 \end{pmatrix}.$$

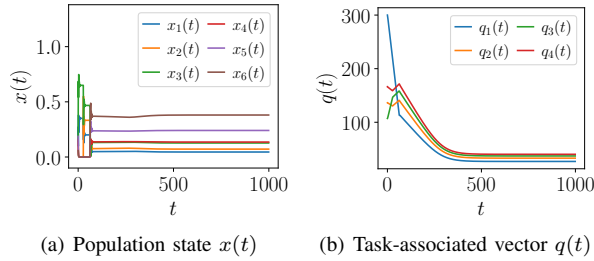


Fig. 5. Graphs depicting the trajectories of the (a) population state $x(t)$ and (b) task-associated vector $q(t)$ derived by the closed-loop model (10) in Example 2 using the growth rate $w = (0.5, 1.0, 1.5, 2.0)$.

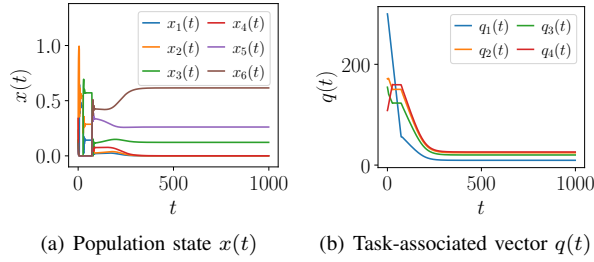


Fig. 6. Graphs depicting the trajectories of the (a) population state $x(t)$ and (b) task-associated vector $q(t)$ derived by the closed-loop model (10) in Example 2 using the growth rate $w = (0.1, 0.5, 1.0, 2.0)$.

ii) For $w = (0.1, 0.5, 1.0, 2.0)$,

$$G = \begin{pmatrix} 1.68 & 0.99 & 0.00 & 0.00 \\ 1.68 & 0.00 & 0.80 & 0.00 \\ 1.68 & 0.00 & 0.00 & 0.67 \\ 0.00 & 0.99 & 0.80 & 0.00 \\ 0.00 & 0.99 & 0.00 & 0.67 \\ 0.00 & 0.00 & 0.80 & 0.67 \end{pmatrix}.$$

By comparing the above two payoff matrices, we can infer that the optimal G assigns higher payoffs to the strategies as their respective growth rates become smaller. Figures 5 and 6 depict the resulting trajectories for $x(t)$ and $q(t)$. Notably, as the growth rate of the 4-th target state becomes relatively higher than those of other target states, more agents adopt strategies 3, 5, and 6, which can be used to measure the 4-th state. Similar to the simulation results in Section IV-B, we can observe a large variation in the agent strategy revision at the beginning of the simulations as the agents responsively revise their strategies based on the value of $q_i(t)$.

V. CONCLUSIONS

We investigated the design of the payoff mechanism in the task allocation games. The mechanism determines payoffs p given the vector q that quantifies the amount of jobs in the assigned tasks to the agents, and the payoffs incentivize the agents to repeatedly revise their strategy selection. We discussed how to design the payoff matrix G using the passivity tools to ensure that the agents asymptotically attain the optimal strategy profile. Using the numerical examples, we demonstrated how our results can be used to design G and how the parameters of the dynamic game model affect the optimal design of G . For future directions, we

plan to consider the design of nonlinear payoff mechanisms $p = G(q)$, and to explore the idea of learning the dynamic model and computing G alongside the model learning.

REFERENCES

- [1] S. Park, Y. D. Zhong, and N. E. Leonard, "Multi-robot task allocation games in dynamically changing environments," in *2021 IEEE International Conference on Robotics and Automation (ICRA)*, 2021.
- [2] L. Parker, "Alliance: an architecture for fault tolerant multirobot cooperation," *IEEE Transactions on Robotics and Automation*, vol. 14, no. 2, pp. 220–240, 1998.
- [3] C. Yang, L. Kaplan, E. Blasch, and M. Bakich, "Optimal placement of heterogeneous sensors for targets with Gaussian priors," *IEEE Transactions on Aerospace and Electronic Systems*, vol. 49, no. 3, pp. 1637–1653, 2013.
- [4] L. Parker and F. Tang, "Building multirobot coalitions through automated task solution synthesis," *Proceedings of the IEEE*, vol. 94, no. 7, pp. 1289–1305, 2006.
- [5] I. Jang, H.-S. Shin, and A. Tsourdos, "Anonymous hedonic game for task allocation in a large-scale multiple agent system," *IEEE Transactions on Robotics*, vol. 34, no. 6, pp. 1534–1548, 2018.
- [6] W. Saad, Z. Han, T. Basar, M. Debbah, and A. Hjørungnes, "Hedonic coalition formation for distributed task allocation among wireless agents," *IEEE Transactions on Mobile Computing*, vol. 10, no. 9, 2011.
- [7] H.-L. Choi, L. Brunet, and J. P. How, "Consensus-based decentralized auctions for robust task allocation," *IEEE Transactions on Robotics*, vol. 25, no. 4, pp. 912–926, 2009.
- [8] S. Amaya and A. Mateus, "Tasks allocation for rescue robotics: A replicator dynamics approach," in *Artificial Intelligence and Soft Computing*, L. Rutkowski, R. Scherer, M. Korytkowski, W. Pedrycz, R. Tadeusiewicz, and J. M. Zurada, Eds. Cham: Springer International Publishing, 2019, pp. 609–621.
- [9] S. Berman, A. Halasz, M. A. Hsieh, and V. Kumar, "Optimized stochastic policies for task allocation in swarms of robots," *IEEE Transactions on Robotics*, vol. 25, no. 4, pp. 927–937, 2009.
- [10] A. Pashaie, L. Pavel, and C. J. Damaren, "A population game approach for dynamic resource allocation problems," *International Journal of Control*, vol. 90, no. 9, pp. 1957–1972, 2017.
- [11] J. R. Marden, "State based potential games," *Automatica*, vol. 48, no. 12, pp. 3075–3088, 2012.
- [12] N. Li and J. R. Marden, "Designing games for distributed optimization," *IEEE Journal of Selected Topics in Signal Processing*, vol. 7, no. 2, pp. 230–242, 2013.
- [13] W. H. Sandholm, *Population Games and Evolutionary Dynamics*. MIT Press, 2011.
- [14] E. Ramirez-Llanos and N. Quijano, "A population dynamics approach for the water distribution problem," *International Journal of Control*, vol. 83, pp. 1947–1964, 2010.
- [15] M. J. Fox and J. S. Shamma, "Population games, stable games, and passivity," *Games*, vol. 4, pp. 561–583, Oct. 2013.
- [16] J. Barreiro-Gomez, C. Ocampo-Martinez, N. Quijano, and J. M. Maestre, "Non-centralized control for flow-based distribution networks: A game-theoretical insight," *Journal of The Franklin Institute*, vol. 354, pp. 5771–5796, 2017.
- [17] S. Park, N. C. Martins, and J. S. Shamma, "From population games to payoff dynamics models: A passivity-based approach," in *2019 IEEE 58th Conference on Decision and Control (CDC)*, 2019.
- [18] S. Park, J. S. Shamma, and N. C. Martins, "Passivity and evolutionary game dynamics," in *2018 IEEE Conference on Decision and Control (CDC)*, 2018, pp. 3553–3560.
- [19] M. Arcak and N. C. Martins, "Dissipativity tools for convergence to Nash equilibria in population games," *IEEE Transactions on Control of Network Systems*, vol. 8, no. 1, pp. 39–50, 2021.
- [20] S. Park and N. E. Leonard, "KL divergence regularized learning model for multi-agent decision making," in *2021 American Control Conference (ACC)*, 2021, pp. 4509–4514.
- [21] —, "Learning with delayed payoffs in population games using Kullback-Leibler divergence regularization," arXiv.org, 2023.
- [22] S. Park and J. Barreiro-Gomez, "Payoff mechanism design for coordination in multi-agent task allocation games," arXiv.org, 2023.