Congestion-Aware Path Coordination Game With Markov Decision Process Dynamics

Sarah H. Q. Li, *Student Member, IEEE*, Daniel Calderone, and Behçet Açıkmeşe, *Fellow, IEEE*

***Abstract*—Inspired by the path coordination problem arising from robo-taxis, warehouse management, and mixed-vehicle routing, we model a group of heterogeneous players responding to stochastic demands as a congestion game under Markov decision process dynamics. Players share a common state-action space but have unique transition dynamics, and each player's unique cost is a *function* of the joint state-action probability distribution. For a class of player cost functions, we formulate the player-specific optimization problem, prove equivalence between the Nash equilibrium and the solution of a potential minimization problem, and derive dynamic programming approaches to solve the Nash equilibrium. We apply this game to model multi-agent path coordination and introduce congestion-based cost functions that enable players to complete individual tasks while avoiding congestion with their opponents. Finally, we present a learning algorithm for finding the Nash equilibrium that has linear complexity in the number of players. We demonstrate our game model on a multi-robot warehouse path coordination problem, in which robots autonomously retrieve and deliver packages while avoiding congested paths.**

***Index Terms*—Markov decision process, stochastic games, path planning, congestion games.**

I. Introduction

As autonomous path planning algorithms become widely-adapted by aeronautical, robotics, and operational sectors [1], [2], the standard assumption that the operating environment is stationary is no longer sufficient. More likely, autonomous players *share* the operating environment with other players who may have conflicting objectives. While the possibility for multi-agent conflicts has pushed single-agent path planning towards greater emphasis on robust planning and collision avoidance, we believe that the overarching goal should be to consider other players' trajectories and achieve optimality with respect to the *multi-agent dynamics*.

We focus on the scenario where a group of heterogeneous players collectively perform path planning in response to stochastic demands. We are inspired by fleets of robo-taxis fulfilling ride demands while avoiding congestion in traffic [3] and warehouse robots retrieving packages under dynamic arrival rates [4], [5] while avoiding collisions. The common feature in these applications is that the players must plan with respect to a forecasted demand distribution rather than a deterministic demand. We assume that the desirable outcome is a competitive equilibrium. Beyond competitive settings, a competitive equilibrium can be used in cooperative settings to ensure that each player achieves identical costs and each demand is *optimally* fulfilled with respect to other demands, thus ensuring a degree of fairness.

We propose MDP congestion games as a theoretical framework for analyzing the resulting path coordination problem. By leveraging common congestion features in multi-agent path planning, our key contribution is *reducing* the N-player coupled MDP problem to a *single* potential minimization problem. As a result, we can use optimization techniques to analyze the Nash equilibrium as well as apply gradient descent methods to compute it.

Contributions: To address the lack of game-theoretical models for path coordination under MDP dynamics, we propose an MDP congestion game with finite players and heterogeneous player costs and dynamics. We define Bellman equation-type conditions for the Nash equilibrium and provide a necessary and sufficient condition for the existence of a game potential. For a subset of player costs, we show equivalence between the Nash equilibrium and the global solution of the potential minimization problem, and provide sufficient conditions for a unique Nash equilibrium. For multi-player path coordination, we study a class of cost functions that allows players to have different sensitivities to the total congestion and to find congestion-free paths that optimally achieve their individual objectives. Finally, we provide a distributed algorithm that converges to the Nash equilibrium and give rates of its convergence. We demonstrate our model and algorithm on a 2D autonomous warehouse problem where robots retrieve and deliver packages with stochastic arrival times while sharing a common navigation space.

II. Related Work

An MDP congestion game [6] is a stochastic population game and is related to potential mean field games [7], [8] in the discrete time and state-action space [9] and mean field games

Manuscript received 22 March 2022; revised 2 June 2022; accepted 21 June 2022. Date of publication 7 July 2022; date of current version 20 July 2022. This work was supported by NSF under Award CMMI-2105502. Recommended by Senior Editor M. Guay. *(Corresponding author: Sarah H. Q. Li.)*

The authors are with the William E. Boeing Department of Aeronautics and Astronautics, University of Washington, Seattle, WA 98195 USA (e-mail: sarahli@uw.edu; djcal@uw.edu; behcet@uw.edu).

Digital Object Identifier 10.1109/LCSYS.2022.3189323

on graphs [10]. In this letter, we extend our previous framework from continuous populations of identical MDP decision makers [6] to a finite number of heterogenous MDP decision makers. In the continuous population case, MDP congestion games have been analyzed for constraint satisfaction in [11] and sensitivity to hyperparameters in [12].

Model-based multi-agent path planning is typically solved via graph-based searches [13] and mixed integer linear programming [14]. Reinforcement learning is also a viable method for solving multi-agent path planning [1], [15]. In most scenarios, the path planning problem is modeled as an MDP [16], [17]. In particular, [17] adopts a stochastic game model for human-robot collision avoidance, but focuses more on algorithm development than game structure analysis.

III. Heterogeneous MDP Congestion Game

Consider a finite number of players $[N] = \{1, \dots, N\}$ with a *shared finite state-action space* given by $([S], [A])$ in time interval $\mathcal{T} = \{0, 1, \dots, T\}$. Each player i has *individual* time-varying transition probabilities given by $P^i \in \mathbb{R}_+^{TSSA}$, where at time t, $P^i_{ts'sa}$ is the transition probability from state s to state s' using action a satisfying the simplex constraints:

$$\sum_{s'} P^i_{ts'sa} = 1, \ \ \forall (i, t, s, a) \in [N] \times [T] \times [S] \times [A]. \quad (1)$$

State-Action Distribution: At time t, let player i's state be $s^i(t) \in [S]$ and action taken be $a^i(t) \in [A]$, then $x^i_{tsa} = \mathbb{P}[s^i(t) = s, a^i(t) = a]$ is player i's probability of being in state s taking action a at time t. Player i's state-action probability trajectory over time period $\mathcal{T}$ is $x^i \in \mathbb{R}^{(T+1)SA}$, its *state-action distribution*. We use $\mathcal{X}(P^i, z^i_0)$ to denote the set of all feasible state-action distributions under transition dynamics P^i and initial condition $z^i_0 \in \mathbb{R}^S_+$, where $z^i_{0s} = \mathbb{P}[s^i(0) = s]$ is player i's probability of starting in state s.

$$\mathcal{X}(P^i, z^i_0) := \Bigg\{ x^i \in \mathbb{R}_+^{(T+1)SA} \Bigg| \sum_a x^i_{0sa} = z^i_{0s}, \forall s \in [S], \sum_{s',a} P^i_{tss'a} x^i_{(t-1)s'a} = \sum_a x^i_{tsa}, \ \forall (t, s) \in [T] \times [S] \Bigg\}. \quad (2)$$

The *joint state-action distribution* of all players is given by

$$x = (x^1, \dots, x^N) \in \mathbb{R}_+^{N(T+1)SA}. \quad (3)$$

We assume that x is fully observable and may denote it as $x = (x^i, x^{-i})$ where $x^{-i} = (x^j)_{j \in [N]/\{i\}}$.

Player Costs: Similar to stochastic games, the player costs are continuously differentiable *functions* of x: player i incurs a cost $\ell^i_{tsa}(x)$ for taking action a at state s and time t.

$$\ell^i_{tsa} : \mathbb{R}_+^{N(T+1)SA} \mapsto \mathbb{R}, \ \forall (i, t, s, a) \in [N] \times \mathcal{T} \times [S] \times [A]. \quad (4)$$

Compared to stochastic games where player costs are coupled to the opponent policies, (4) is better suited to model collision events. For example, the expectation of the log-barrier function for players i and j at time t can be modeled as $\sum_{s,s' \in [S]} (\sum_a x^i_{tsa})(\sum_a x^j_{ts'a}) \log(d_{s,s'})$, in which $d_{s,s'}$ denotes the distance between states $s, s' \in [S]$.

The *cost vector* of $(\ell^1, \dots \ell^N)$ (4) is given by $\xi : \mathbb{R}_+^{N(T+1)SA} \mapsto \mathbb{R}_+^{N(T+1)SA}$,

$$\xi(x) = [\ell^1_{011}(x), \ell^1_{012}(x), \dots, \ell^N_{TSA}(x)] \in \mathbb{R}_+^{N(T+1)SA}. \quad (5)$$

We assume that ξ has a positive definite gradient in x.

Assumption 1: The player cost vector ξ (5) satisfies $\nabla \xi(x) \succ 0$ for all x (3) where $x^i \in \mathcal{X}(P^i, x^i_0), \ \forall i \in [N]$.

For the class of player costs considered in Section III-B, Assumption 1 implies that the player costs strictly increase as the number of players increases.

Coupled MDPs: Given an initial distribution $z^i_0 \in \mathbb{R}^S_+$ and fixed state-action distributions x^{-i} (3), player i solves the following optimization problem under MDP dynamics.

$$\min_{x^i} \sum_{t,s,a} \int_0^{x^i_{tsa}} \ell^i_{tsa}(u^i, x^{-i}) \partial u^i_{tsa} \text{ s.t. } x^i \in \mathcal{X}(P^i, z^i_0). \quad (6)$$

In (6), each integral is taken over u^i_{tsa}, the $(t, s, a)^{th}$ element of u^i. When $\ell^i_{tsa}(x)$ is constant for all $(t, s, a) \in \mathcal{T} \times [S] \times [A]$, player i solves a standard *linear program* MDP.

Dynamic Programming: At a joint state-action distribution x (3), player i's cost-to-go in (6) can be recursively defined via Q-value functions [18] as

$$\begin{aligned} Q^i_{Tsa}(x) &:= \ell^i_{Tsa}(x), \\ Q^i_{(t-1)sa}(x) &:= \ell^i_{(t-1)sa}(x) + \sum_{s'} P^i_{ts'sa} \min_{a'} Q^i_{t,s'a'}(x), \\ &\forall \ t \in [T] \end{aligned} \quad (7)$$

The optimal solution of (6) can be stated using (7).

Theorem 1: Under Assumption 1, x^i (2) uniquely minimizes (6) with respect to the state-action distribution x^{-i} if and only if its associated $Q^i(x^i, x^{-i})$ (7) satisfies

$$x^i_{tsa} > 0 \Rightarrow Q^i_{tsa}(x^i, x^{-i}) = \min_{a'} Q^i_{tsa'}(x^i, x^{-i}), \quad (8)$$

for all $(t, s, a) \in \mathcal{T} \times [S] \times [A]$. I.e., x^i is optimal for (6) if and only if every action played with nonzero probability achieves the minimum cost-to-go (7) among available actions.

Proof: Let $F(x^i, x^{-i}) = \sum_{t,s,a} \int_0^{x^i_{tsa}} \ell^i_{tsa}(u^i, x^{-i}) \partial u^i_{tsa}$, then $\partial F(x^i, x^{-i}) / \partial x^i = \ell(x^i, x^{-i})$. We then apply Proposition A1 to (6) and the theorem's results follow directly. ■

When all players jointly achieve the optimal cost-to-go (8), a stable equilibrium for unilateral optimality is achieved.

Definition 1 (Nash Equilibrium): The joint state-action distribution $\hat{x} = [\hat{x}^1, \dots, \hat{x}^N]$ (3) is a *Nash equilibrium* if $(\hat{x}^i, Q^i(\hat{x}))$ satisfies (8) for all $i \in [N]$.

A. Potential Optimization Form

We are interested in MDP congestion games that can be reduced from the coupled MDPs (6) to a single minimization problem given by

$$\min_{x^1, \dots, x^N} F(x), \ \text{ s.t. } x^i \in \mathcal{X}(P^i, z^i_0), \ \forall \ i \in [N], \quad (9)$$

where F is the *potential function* of the corresponding game.

Definition 2 (Potential Function): We say an MDP congestion game with player costs $\{\ell^i\}_{i\in[N]}$ (4) has a *potential function* $F{:}\mathbb{R}^{N(T+1)SA} \mapsto \mathbb{R}$ if F satisfies

$$\frac{\partial F(x)}{\partial x^i_{tsa}} = \ell^i_{tsa}(x), \ \ \forall\ (i,t,s,a) \in [N]\times\mathcal{T}\times[S]\times[A]. \quad (10)$$

The following assumption on $\{\ell^i\}_{i\in[N]}$ is necessary and sufficient for the existence of F [19, eqn. 2.44].

Assumption 2: For all $(i,t,s,a), (i',t',s',a') \in [N]\times\mathcal{T}\times[S]\times[A]$, the player costs $\{\ell^i\}_{i\in[N]}$ satisfy

$$\frac{\partial \ell^i_{tsa}(x)}{\partial x^{i'}_{t's'a'}} = \frac{\partial \ell^{i'}_{t's'a'}(x)}{\partial x^i_{tsa}}. \quad (11)$$

Remark 1: Assumption 2 is equivalent to F being conservative: $\forall\ x_1, x_2 \in \{x^i_{tsa} \mid (i,t,s,a) \in [N]\times\mathcal{T}\times[S]\times[A]\}$,

$$\partial^2 F(x)/\partial x_1 \partial x_2 = \partial^2 F(x)/\partial x_2 \partial x_1. \quad (12)$$

In other words, the Jacobian of ξ (5), $\partial\xi(x)/\partial x$, is symmetrical.

Verifying the existence of F (10) is non-trivial. However, if F exists, the solution of (9) is the Nash equilibrium [20].

Theorem 2: If the player costs $\{\ell^i\}_{i\in[N]}$ (4) satisfy Assumption 1,

1) the potential function (Definition 2) exists,
2) $\hat{x}$ (3) is the global optimal solution of (9) if and only if $\hat{x}$ is a Nash equilibrium (Definition 1).

Proof: We prove statement 1 by showing that Assumption 1 implies Assumption 2: if $\nabla\xi(x) \succ 0$ for all feasible joint state-action distributions x (3), then $\nabla\xi(x)$ is symmetrical and satisfies (11). Next, we show the forward direction of statement 2. If $(\hat{x}^1, \dots \hat{x}^N)$ minimizes (9), then for each $i \in [N]$, $\hat{x}^i$ minimizes (22) at $\hat{x}^{-i}$. From Proposition A1, $\hat{x}^i$ satisfies (8) for all $i \in [N]$, therefore $\hat{x}$ is a Nash equilibrium. To show the reverse direction of 2, if (8) is satisfied for all $i \in [N]$, $\hat{x}^i$ is coordinate-wise optimal for coordinate i (Proposition A1). Under Assumption 1, (9) has a strictly convex differentiable objective with separable convex constraints $\mathcal{X}(P^i, z^i_0)$—each x^i is constrained independently of x^j, $\forall j \in [N]/\{i\}$, then the jointly coordinate-wise optimal $\hat{x}$ is the global optimal solution of (9) [21, Th. 4.1]. ∎

B. Path Coordination as an MDP Congestion Game

We now model the path coordination problem as an MDP congestion game and demonstrate how players can achieve individual objectives while avoiding each other.

To reflect the congestion level of each state-action, we first define a **congestion distribution** as the weighted sum of individual state-action distributions.

$$y := \sum_{i\in[N]} \alpha_i x^i \in \mathbb{R}^{(T+1)SA}, \ \ \alpha_i > 0, \ \ \forall i \in [N], \quad (13)$$

where α_i is player i's *impact factor*. If all players contribute to congestion equally, $\alpha_i = 1\ \forall i \in [N]$.

Player Costs: We derive a class of player costs that satisfy Assumption 1, incorporate congestion-based penalties, and enable players to pursue individual objectives. For all $(i,t,s,a) \in [N]\times\mathcal{T}\times[S]\times[A]$, the player cost is given by

$$\ell^i_{tsa}(y, x^i) = \alpha_i f_{ts}\left(\sum_{a'} y_{tsa'}\right) + \alpha_i g_{tsa}(y_{tsa}) + h^i_{tsa}(x^i_{tsa}), \quad (14)$$

where α_i is the same as in (13), $f_{ts} : \mathbb{R} \mapsto \mathbb{R}$ is the state-dependent congestion and takes the congestion level of (t,s) as input, $g_{tsa} : \mathbb{R} \mapsto \mathbb{R}$ is the state-action-dependent congestion and takes the congestion level of (t,s,a) as input, and $h^i_{tsa} : \mathbb{R} \mapsto \mathbb{R}$ is the player-specific objective and takes player i's probability of being in (t,s,a) as input. Player-specific objectives such as obstacle avoidance and target reachability can be incorporated as constant offsets in h^i.

Remark 2 (Effect of α_i): The impact factor α_i scales player i's relative impact on the total congestion and the total congestion's impact on player i. When $\alpha_i < \alpha_j$, player i impacts congestion less and cares about the congestion less than player j. When $\alpha_i > \alpha_j$, player i impacts congestion more and cares about the congestion more than player j.

The potential function (10) of the game with costs (14) is

$$F(x) = \sum_{t,s}\int_0^{\sum_{a'} y_{tsa'}} f_{ts}(u)\partial u + \sum_{t,s,a}\int_0^{y_{tsa}} g_{tsa}(u)\partial u + \sum_{i,t,s,a}\int_0^{x^i_{tsa}} h^i_{tsa}(u)\partial u. \quad (15)$$

Remark 3: Congestion costs f and g must be identical for all players in order for a potential (Definition 2) to exist.

Example 1 (Road-Sharing Vehicles): Consider a sedan (player 1, $\alpha_1 = 1$) and a trailer (player 2, $\alpha_2 = 2$) sharing a road network modeled by $[S]\times[A]$. Player i wants to reach state $s_i \in [S]$. The player-specific objective is $h^i_{tsa}(x^i_{tsa}) = -\mathbb{1}[s = s_i] + \epsilon_i x^i_{tsa}$, where $\mathbb{1}[w]$ is 1 when w is true and 0 otherwise. The term $\epsilon_i x^i_{tsa}$ where $\epsilon_i > 0$ encourages player i to randomize its policy over all optimal actions. Players experience state-based congestion as $f_{ts}(w) = \exp(w)$. The player cost (14) is $\ell^i_{tsa}(y, x^i) = \alpha_i \exp(\sum_{a'} y_{tsa'}) + \epsilon_i x^i_{tsa} - \mathbb{1}[s = s_i]$.

Corollary 1: Player costs of form (14) satisfy Assumption 1 if $h^i_{tsa}(\cdot)$ is strictly increasing and $f_{ts}(\cdot)$, $g_{tsa}(\cdot)$ are non-decreasing $\forall (i,t,s,a) \in [N]\times\mathcal{T}\times[S]\times[A]$.

Proof: Let I_Z be an identity matrix of size $Z\times Z$, $\mathbf{1}_Z$ be a ones vector of size $Z\times 1$, $\vec{\alpha} = [\alpha_1, \dots, \alpha_N] \in \mathbb{R}^{N\times 1}$, $h(x) = [h^1(x), \dots, h^N(x)] \in \mathbb{R}^{N(T+1)SA}$, and $\otimes$ be a Kronecker product. We define the matrices $M = \vec{\alpha} \otimes I_{(T+1)SA}$ and $J = (I_{(T+1)S} \otimes \mathbf{1}_A^\top)M$, and verify that $Mx = y$, $[Jx]_{ts} = \sum_{a'} y_{tsa'}$ $\forall (t,s) \in \mathcal{T}\times[S]$, and $\xi(x) = J^\top f(Jx) + M^\top g(Mx) + h(x)$. Let $w = Jx$, we can take ξ's gradient as $\nabla\xi(x) = J^\top \nabla f(w) J + M^\top \nabla g(y) M + \nabla h(x)$. Under Corollary assumptions, $\nabla f(w)$ and $\nabla g(y)$ are non-negative diagonal matrices and $\nabla h(x)$ is a strictly positive diagonal matrix. Therefore, $\nabla\xi(x) \succ 0$. ∎

Remark 4: Corollary 1 implies that a strictly increasing h^i is crucial to ensuring a unique Nash equilibrium. Therefore, h^i can be interpreted as a regularization term.

C. Frank-Wolfe Learning Dynamics

We find the Nash equilibrium of MDP congestion games by leveraging single-agent dynamic programming.

Algorithm 1 Frank-Wolfe With Dynamic Programming

Require: $\{\ell^i\}_{i\in[N]}$, $\{P^i\}_{i\in[N]}$, $\{z_0^i\}_{i\in[N]}$, N, $[S]$, $[A]$, $\mathcal{T}$.
Ensure: $\{\hat{x}^i_{tsa}\}_{t\in\mathcal{T},s\in[S],a\in[A]}$.
1: $x^{i0} \in \mathcal{X}(P^i, z_0^i) \in \mathbb{R}^{(T+1)SA}, \quad \forall\, i \in [N]$.
2: **for** $k = 1, 2, \ldots,$ **do**
3: **for** $i = 1, \ldots, N$ **do**
4: $C^{ik} = \ell^i([x^{1k}, \ldots, x^{Nk}])$
5: $\pi^i = \text{MDP}(C^{ik}, P^i, [S], [A], T, z_0^i)$
6: $b^{ik} = \text{RETRIEVEDENSITY}(P, z_0^i, \pi^i)$ ▷ Alg. 2
7: $x^{i(k+1)} = \left(1 - \frac{2}{k+1}\right)x^{ik} + \frac{2}{k+1}b^{ik}$
8: **end for**
9: **end for**

Algorithm 2 Retrieving State-Action Distribution From π

Require: P, z, π.
Ensure: $\{d_{tsa}\}_{t\in\mathcal{T},s\in[S],a\in[A]}$
1: $d_{0s\pi_{0s}} = z_s, \ \forall s \in [S]$
2: **for** $t = 1, \ldots, T$ **do**
3: $d_{ts(\pi_{ts})} = \sum_a \sum_{s'} P_{tss'a} d_{(t-1)s'a}, \ \forall\, s \in [S]$
4: **end for**

In Algorithm 1, players access an *oracle* that evaluates the player costs. In line 5, $\pi^i \in [A]^{(T+1)S}$ is any optimal deterministic policy for the finite time MDP with cost C^{ik}, transition probability P^i, and initial distribution z_0^i. We use value iteration to recursively find π^i as

$$\begin{aligned} V^i_{Ts} &= \min_a C^{ik}_{Tsa}, \ \ \pi^i_{Ts} \in \operatorname*{argmin}_a C^{ik}_{Tsa}, \\ V^i_{(t-1)s} &= \min_a C^{ik}_{(t-1)sa} + \sum_{s'} P^i_{ts'sa} V^i_{ts'} \ \ \forall t \in [T] \\ \pi^i_{(t-1)s} &\in \operatorname*{argmin}_a C^{ik}_{(t-1)sa} + \sum_{s'} P^i_{ts'sa} V^i_{ts'} \ \ \forall t \in [T] \end{aligned} \tag{16}$$

Algorithm 1 then retrieves the corresponding state-action density b^{ik} via Algorithm 2 and combines it with the current state-action density x^{ik} to derive the next joint state-action density. All steps within lines 4 to 7 are parallelizable.

Theorem 3: Under Assumption 1, Algorithm 1 converges towards the Nash equilibrium $\hat{x} = (\hat{x}^1, \ldots, \hat{x}^N)$ as

$$\frac{\alpha}{2} \sum_{i\in[N]} \left\| x^{ik} - \hat{x}^i \right\|_2^2 \leq \frac{2C_F}{k+2} \tag{17}$$

where C_F is the potential function F's (10) *curvature constant* given by

$$C_F = \sup_{\substack{x^i, s^i \in \mathcal{X}(P^i, z_0^i) \\ \gamma \in [0,1] \\ w^i = x^i + \gamma(s^i - x^i)}} \frac{2}{\gamma^2} \left(F(s) - F(x) - \sum_{i\in[N]} (x^i - w^i)^\top \ell^i(x) \right).$$

Proof: Algorithm 1 is a straight-forward implementation of [22, Algorithm 2]. From Assumption 1, $\nabla\xi(\hat{x}) \succ 0$. Therefore, the potential function F is strongly convex and satisfies $\frac{\alpha}{2}\sum_{i\in[N]} \left\| x^{ik} - \hat{x}^i \right\|_2^2 \leq F(x^k) - F(\hat{x})$. Equation (17) then follows directly from [22, Th. 1]. ■

Remark 5 (Scalability): Algorithm 1 has linear complexity in the number of players.

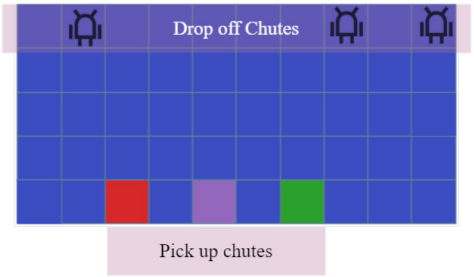

Fig. 1. Operation environment for multi-robot warehouse scenario.

IV. Multi-Agent Path Coordination

We apply our game model to a multi-agent pick up and delivery scenario with stochastic package arrival times. As shown in Figure 1, N players navigate a 2D space. Each player's goal is to transport packages from the pick up chutes to the drop off chutes while avoiding collision with others. Code for the simulation is available at https://github.com/lisarah/mdp_path_coordination.

A. Stationary MDP Model

Players operate in a two dimensional grid world with 5 rows and 10 columns. In addition to capturing location, each state also dictates whether the robot is in pick up or delivery mode. The state space is given by

$$[S] = \{(v, w, m) | 1 \leq v \leq 5, \ \ 1 \leq w \leq 10, \ \ m \in \{1, 2\}\}.$$

At each state, available actions are $[A] = \{u, d, r, l, s\}$, corresponding to up, down, right, left, stay. Player transition dynamics and rewards are *stationary* in time. The transition probability of each state (v, w, m) extends the location-based transition probabilities P^0.

Location-Based Transition: Let $u = (v, w)$ denote the location component of the state. At each location, each action either points to a feasible target $u_{targ}(a)$ or is infeasible. The set of all feasible targets from u is $\mathcal{N}(u)$. When a target exists, players have $1 > q > 0$ chance of reaching it and $1-q$ chance of reaching other states in $\mathcal{N}(u)$.

$$P^0_{u'ua} = \begin{cases} q & u' = u_{targ}(a), \\ \frac{1-q}{|\mathcal{N}(u)|} & u' \in \mathcal{N}(u)/\{u_{targ}(a)\}, \\ 0 & \text{otherwise.} \end{cases} \tag{18}$$

When the target location is infeasible, the player transitions into a neighboring state $u' \in \mathcal{N}(u)$ at random.

$$P^0_{u'ua} = \begin{cases} \frac{1}{|\mathcal{N}(u)|} & u' \in \mathcal{N}(u), \\ 0 & \text{otherwise.} \end{cases} \tag{19}$$

Full Transition Dynamics: Within the same mode, players transition between locations via dynamics P^0. Player modes transition at pick up chutes $\mathcal{P}$ and drop off chutes $\mathcal{D}$.

1) When player i is in mode 1 (pick up) and about to transition into $p^i \in \mathcal{P}$, player i's mode has r^i probability of switching to mode 2 (drop off).

$$\begin{cases} P^i_{t(p^i,2)sa} = r^i P^0_{tp^iua}, \\ P^i_{t(p^i,1)sa} = (1 - r^i) P^0_{tp^iua}, \end{cases} \quad \forall s = (u, 1), \ \ s \in [S].$$

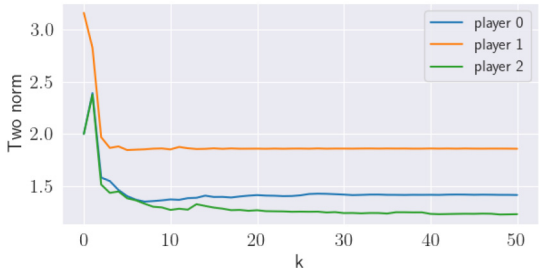

Fig. 2. $\|\cdot\|_2$ of player i's state-action distribution over Algorithm 1 iterations.

2) When player i is in mode 2 and about to transition into $d^i \in \mathcal{D}$, player i switches to mode 1 with probability 1.

$$\begin{cases} P^i_{t(d^i,1)sa} = P^0_{td^iua'} & \forall s = (u,2),\ s \in [S]. \\ P^i_{t(p^i,2)sa} = 0, \end{cases}$$

Here, $r^i \in \mathbb{R}$ denotes the probability of package arrival when player i is in p^i. Modeled as an independent Poisson process with rate λ_i and interval $\Delta t = 1s$, $r^i = \exp(-\lambda_i \Delta_t)$.

B. Player Costs

For all $(t, s, a) \in \mathcal{T} \times [S] \times [A]$ and congestion distribution y (13), player i's cost is given by

$$\ell^i_{tsa}(y, x^i) = \epsilon x^i_{tsa} - c^i_{tsa} + \alpha_i f_{ts}(y).$$

The player-specific objective c^i_{tsa} is defined as

$$c^i_{t(v,w,m)a} = \begin{cases} 1 & (v,w) = p^i,\ m = 1, \\ 1 & (v,w) = d^i,\ m = 2, \\ 0 & \text{otherwise.} \end{cases} \tag{20}$$

The congestion function is strictly state-based and is an exponential function given by

$$f_{t(v,w,m)}(y) = -\beta \exp\left(\beta\Big(\sum_{m' \in \{1,2\}} \sum_{a' \in [A]} y_{t(v,w,m')a'} - 1\Big)\right), \tag{21}$$

where $\alpha_i > 0$ for all $(t, s, a) \in \mathcal{T} \times [S] \times [A]$. As opposed to (14), function (21) calculates the congestion in $(v, w, \cdot)$ using both $(v, w, 1)$'s and $(v, w, 2)$'s congestion level.

C. Simulation Results

We simulate the path coordination game using parameters from Table I. Player i's pick up locations is the i^{th} element of $\mathcal{P} = \{(4, w^i) | w^i \in [8, 7, 2]\}$, and its drop-off location is the i^{th} element of $\mathcal{D} = \{(0, w^i) | w^i \in [4, 5, 8]\}$. At $t = 0$, players are initialized at their drop off location.

We run Algorithm 1 for 100 iterations, where line 5 is solved via value iteration (16). The two norm of x^i is shown in Figure 2 as a function of the algorithm iterations, where the state-action densities stabilize in about 20 steps. Performance is evaluated by: 1) expected number of collisions, 2) expected packages delivery time, 3) worst package delivery time. The results for 100 random trials are visualized in Figures 3 and 4.

We compare the *jointly optimal congestion-free wait time* computed using Algorithm 1 to the shortest wait time available in the absence of opponents. Each path is the number of steps to complete the drop off-pick up-drop off cycle. Based on the pick-up and drop-off locations, each player's shortest wait time without opponents is 16, 12, 20 respectively. This matches well with the average wait time shown in Figure 4.

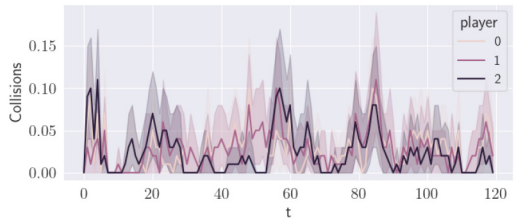

Fig. 3. Collisions per player as a function of MDP time step t.

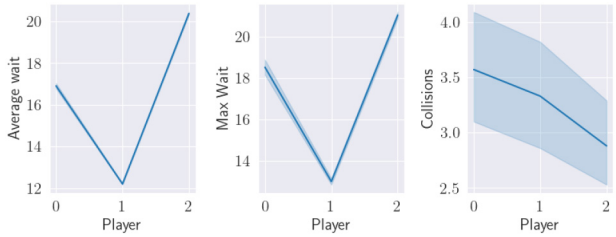

Fig. 4. Average waiting time per package, worst case waiting time per package, and average number of collisions in T for each player.

TABLE I
PARAMETERS FOR SIMULATION ENVIRONMENT

N	q	γ_i	λ_i	α_i	Δt	T	ϵ	β
3	0.98	0.99	0.5	{0.5, 1, 1.5}	1s	120s	1e-3	40

We set the player impact factors as {0.5, 1, 1.5} as in Table I. From Figure 4, the impact factors directly correlate with the rate of collision players experience. Player 0 impacts congestion the least and is the least sensitive to congestion. As a result, it encountered the most collisions. Player 2 impacts congestion the most and is the most sensitive to congestion. As a result, it encountered the least collisions. The collision rate is spread out evenly over $\mathcal{T}$ (Figure 3).

V. CONCLUSION

We derived a class of N player, weighted potential games under heterogeneous MDP dynamics and applied it to multi-agent path coordination. For these games, we showed equivalence between the unique Nash equilibrium and the global solution of a potential minimization problem, which we solved via gradient descent and single-player dynamic programming. Future work includes deriving learning-based solutions for the games and integrating partially observable scenarios in which players have local observations only.

APPENDIX

Proposition 1: Under Assumption 1, consider the problem

$$\min_{x^i} F(x^i, x^{-i}) \ \text{ s.t. } x^i \in \mathcal{X}(P^i, z^i_0). \tag{22}$$

where for $i \in [N]$ and x^{-i}, the objective $F : \mathbb{R}^{N(T+1)SA} \mapsto \mathbb{R}$ satisfies $\partial F(x^i, x^{-i})/\partial x^i = \ell^i(x^i, x^{-i})\ \forall x^i \in \mathcal{X}(P^i, z_0^i)$, then $\hat{x}^i$ minimizes (22) if and only if $Q^i(\hat{x}^i, x^{-i})$ in (7) satisfies (8).

Proof: Because (22) has linear constraints and $\partial^2 F(x)/\partial x_i^2 = \partial \ell^i(x)/\partial x_i \succ 0$ by assumption, (22)'s unique minimizer satisfies the first order KKT conditions. Consider the dual variables $\mu^i \in \mathbb{R}_+^{(T+1)SA}$ for $x^i \geq 0$ and $\nu^i \in \mathbb{R}^{(T+1)SA}$ for the equality constraints in $\mathcal{X}(P^i, z_0^i)$ (2). The Lagrangian of (22) is $L(x^i, \nu^i, \mu^i) = F(x^i, x^{-i}) - \sum_{t,s,a} \mu^i_{tsa} x^i_{tsa} + \sum_s \nu^i_{0s}(x^i_{0s} - \sum_a x^i_{0sa}) + \sum_{s,t} \nu^i_{ts}(\sum_{s'a} P^i_{tss'a} x^i_{(t-1)sa} - \sum_a x^i_{tsa})$. The KKT conditions are 1) primal feasibility $x^i \in \mathcal{X}(P^i, z_0^i)$, 2) dual feasibility $\mu^i \geq 0$, 3) complementary slackness $\mu^i_{tsa} x^i_{tsa} = 0$, $\forall (t, s, a) \in \mathcal{T} \times [S] \times [A]$, and 4) stationarity condition, given $\forall (t, s, a) \in \mathcal{T} \times [S] \times [A]$ as

$$\begin{cases} \ell^i_{tsa}(x) + \sum_{s'} P^i_{(t+1)s'sa} \nu^i_{(t+1)s'} = \nu^i_{ts} + \mu^i_{tsa} & t \neq T, \\ \ell^i_{Tsa}(x) = \nu^i_{Ts} + \mu^i_{Tsa} & t = T. \end{cases} \tag{23}$$

We can show that $(\hat{x}^i, x^{-i})$ satisfies the KKT conditions above if and only if it satisfies (8). To simplify notation, we use Q^i_{tsa} to denote $Q^i_{tsa}(\hat{x}^i, x^{-i})$.

(⇒): suppose $(\hat{x}^i, \nu^i, \mu^i)$ satisfies the KKT conditions. When $\hat{x}^i_{tsa} > 0$, ν^i_{ts} represents the value function and $\nu^i_{ts} + \mu^i_{tsa}$ represents Q-value. When $\hat{x}^i_{tsa} = 0$, we *shift* (ν^i, μ^i) to $(\hat{\nu}^i, \hat{\mu}^i)$ to generate the optimal Q-values. To this end, define $\lambda^i \in \mathbb{R}^{(T+1)SA}$, $\Delta^i \in \mathbb{R}^{(T+1)S}$, $\hat{\mu}^i \in \mathbb{R}^{(T+1)SA}$, $\hat{\nu}^i \in \mathbb{R}^{(T+1)S}$ recursively from $t = T$. At $T = t$, let $\Delta^i_{(T+1)s'} = 0\ \forall s' \in [S]$. All other variables are recursively defined as

$$\begin{aligned} \lambda^i_{tsa} &= \mu^i_{tsa} + \sum_{s'} P^i_{(t+1)s'sa} \Delta^i_{(t+1)s'}, \\ \Delta^i_{ts} &= \min_{a'} \lambda^i_{tsa'}, \\ \hat{\mu}^i_{tsa} &= \lambda^i_{tsa} - \Delta^i_{ts}, \\ \hat{\nu}^i_{ts} &= \nu^i_{ts} + \Delta^i_{ts}. \end{aligned} \tag{24}$$

At time t, let the condition $\hat{x}^i_{tsa} > 0$ implies $\lambda^i_{tsa} = 0$ be denoted as $\mathcal{K}(t)$. We can show that $\mathcal{K}(t)$ implies $\mathcal{K}(t-1)$: from complementary slackness, $\hat{x}^i_{(t-1)sa} > 0$ implies $\mu^i_{(t-1)sa} = 0$. Subsequently, $\lambda^i_{(t-1)sa} = 0$ (24) if $P^i_{ts'sa} \Delta^i_{ts'} = 0\ \forall s' \in [S]$: either $P^i_{ts'sa} = 0$ or $P^i_{ts'sa} \hat{x}^i_{(t-1)sa} = \sum_{a'} \hat{x}^i_{ts'a'} > 0$. In the second case, there exists $a' \in [A]$ such that $\hat{x}^i_{ts'a'} > 0$, and if $\mathcal{K}(t)$ holds, $\lambda^i_{ts'a'} = 0$. By definition, $\Delta^i_{ts'}$ is non-negative and must be zero. We conclude that $P^i_{ts'sa} \Delta^i_{ts'} = 0\ \forall s' \in [S]$, and $\mathcal{K}(t-1)$ holds. At $t = T$, $\hat{x}^i_{Tsa} > 0$ implies $\mu^i_{Tsa} = 0$ and $\lambda^i_{Tsa} = 0$. Therefore, $\mathcal{K}(t)$ holds $\forall t \in \mathcal{T}$.

By adding $\sum_{s'} P^i_{(t+1)s'sa} \Delta^i_{(t+1)s'}$ to (23) and simplifying it via (24), we obtain

$$\begin{cases} \ell^i_{tsa}(x) + \sum_{s'} P^i_{(t+1)s'sa} \hat{\nu}^i_{(t+1)s} = \hat{\nu}^i_{ts} + \hat{\mu}^i_{tsa} & t \neq T \\ \ell^i_{Tsa}(x) = \hat{\nu}^i_{Ts} + \hat{\mu}^i_{Tsa} & t = T. \end{cases} \tag{25}$$

We define $Q^i_{tsa} = \hat{\nu}^i_{ts} + \hat{\mu}^i_{tsa}$. From (24), $\hat{\mu}^i_{tsa}$ is always non-negative and $\hat{\mu}^i_{tsa'} = 0$ for some $a' \in [A]$. Therefore $\min_{a'} Q^i_{tsa'} = \hat{\nu}^i_{ts}$, and Q^i substituted in (25) satisfies (7).

If $\hat{x}^i_{tsa} > 0$, then from $\mathcal{K}(t)$, $\lambda^i_{tsa} = 0$. Therefore, $\hat{\mu}^i_{tsa} = 0$ and $Q^i_{tsa} = \min_{a'} Q^i_{tsa'}$. We conclude that Q^i satisfies (8).

(⇐): we show that if Q^i satisfies (8), then $\hat{x}^i$ satisfies the KKT conditions. Let $\nu^i_{ts} = \min_{a'} Q^i_{tsa'}$ and $\mu^i_{tsa} = Q^i_{tsa} - \nu^i_{ts}$ $\forall (t, s, a) \in \mathcal{T} \times [S] \times [A]$, then $(\hat{x}^i, \nu^i, \mu^i)$ is a KKT point. Both $\hat{x}^i$ and μ^i satisfy primal/dual feasibility respectively. From (8), $\hat{x}^i_{tsa} > 0$ implies that $\nu^i_{ts} = Q^i_{tsa}$ and $\mu^i_{tsa} = 0$. Since either $\hat{x}^i_{tsa} > 0$ or $\hat{x}^i_{tsa} = 0$, complementary slackness $\hat{x}^i_{tsa} \mu^i_{tsa} = 0$ holds $\forall (t, s, a) \in \mathcal{T} \times [S] \times [A]$. Finally, the stationarity condition (23) directly follows from (7). ■

REFERENCES

[1] K. Yun *et al.*, "Multi-agent motion planning using deep learning for space applications," in *Proc. ASCEND*, 2020, p. 4233.

[2] J. Ota, "Multi-agent robot systems as distributed autonomous systems," *Adv. Eng. Inform.*, vol. 20, no. 1, pp. 59–70, 2006.

[3] R. Vosooghi, J. Kamel, J. Puchinger, V. Leblond, and M. Jankovic, "Robo-taxi service fleet sizing: Assessing the impact of user trust and willingness-to-use," *Transport*, vol. 46, no. 6, pp. 1997–2015, 2019.

[4] N. V. Kumar and C. S. Kumar, "Development of collision free path planning algorithm for warehouse mobile robot," *Procedia Comput. Sci.*, vol. 133, pp. 456–463, Jan. 2018.

[5] Z. Li, A. V. Barenji, J. Jiang, R. Y. Zhong, and G. Xu, "A mechanism for scheduling multi robot intelligent warehouse system face with dynamic demand," *J. Intell. Manuf.*, vol. 31, no. 2, pp. 469–480, 2020.

[6] D. Calderone and S. Shankar, "Infinite-horizon average-cost Markov decision process routing games," in *Proc. Intell. Transp. Syst.*, 2017, pp. 1–6.

[7] J.-M. Lasry and P.-L. Lions, "Mean field games," *Jpn. J. Math.*, vol. 2, no. 1, pp. 229–260, 2007.

[8] O. Guéant, "From infinity to one: The reduction of some mean field games to a global control problem," 2011, *arXiv:1110.3441*.

[9] D. A. Gomes, J. Mohr, and R. R. Souza, "Discrete time, finite state space mean field games," *J. Math. Pures Appl.*, vol. 93, no. 3, pp. 308–328, 2010.

[10] O. Guéant, "Existence and uniqueness result for mean field games with congestion effect on graphs," *Appl. Math. Optim.*, vol. 72, no. 2, pp. 291–303, 2015.

[11] S. H. Li, Y. Yu, D. Calderone, L. Ratliff, and B. Açıkmeşe, "Tolling for constraint satisfaction in Markov decision process congestion games," in *Proc. Amer. Control Conf. (ACC)*, 2019, pp. 1238–1243.

[12] S. H. Li, D. Calderone, L. Ratliff, and B. Açıkmeşe, "Sensitivity analysis for Markov decision process congestion games," in *Proc. Conf. Decis. Control (CDC)*, 2019, pp. 1301–1306.

[13] L. Cohen, T. Uras, T. S. Kumar, and S. Koenig, "Optimal and bounded-suboptimal multi-agent motion planning," in *Proc. Annu. Symp. Combinatorial Search*, 2019, pp. 44–51.

[14] J. Chen, J. Li, C. Fan, and B. Williams, "Scalable and safe multi-agent motion planning with nonlinear dynamics and bounded disturbances," 2020, *arXiv:2012.09052*.

[15] S. H. Semnani, H. Liu, M. Everett, A. De Ruiter, and J. P. How, "Multi-agent motion planning for dense and dynamic environments via deep reinforcement learning," *IEEE Robot. Autom. Lett.*, vol. 5, no. 2, pp. 3221–3226, Apr. 2020.

[16] H. Bayerlein, M. Theile, M. Caccamo, and D. Gesbert, "Multi-UAV path planning for wireless data harvesting with deep reinforcement learning," *IEEE Open J. Commun. Soc.*, vol. 2, pp. 1171–1187, 2021.

[17] S.-Y. Lo, B. Fernandez, P. Stone, and A. L. Thomaz, "Towards safe motion planning in human workspaces: A robust multi-agent approach," in *Proc. Int. Conf. Robot. Autom. (ICRA)*, 2021, pp. 7929–7935.

[18] M. L. Puterman, *Markov Decision Processes: Discrete Stochastic Dynamic Programming*. New York, NY, USA: Wiley, 2014.

[19] M. Patriksson, *The Traffic Assignment Problem: Models and Methods*. Utrecht, The Netherlands: Courier Dover Publ., 2015.

[20] D. Calderone and S. S. Sastry, "Markov decision process routing games," in *Proc. Int. Conf. Cyber-Phys. Syst.*, 2017, pp. 273–280.

[21] P. Tseng, "Convergence of a block coordinate descent method for non-differentiable minimization," *J. Optim. Theory Appl.*, vol. 109, no. 3, pp. 475–494, 2001.

[22] M. Jaggi, "Revisiting Frank-Wolfe: Projection-free sparse convex optimization," in *Proc. Int. Conf. Mach. Learn.*, 2013, pp. 427–435.