Intention Communication and Hypothesis Likelihood in Game-Theoretic Motion Planning

Makram Chahine[1], Roya Firoozi[2], Wei Xiao[1], Mac Schwager[2] and Daniela Rus[1]

Abstract—Game-theoretic motion planners are a potent solution for controlling systems of multiple highly interactive robots. Most existing game-theoretic planners unrealistically assume *a priori* objective function knowledge is available to all agents. To address this, we propose a fault-tolerant receding horizon game-theoretic motion planner that leverages inter-agent communication with intention hypothesis likelihood. Specifically, robots communicate their objective function which incorporates their intentions. A discrete Bayesian filter is designed to infer the objectives in real-time based on the discrepancy between observed trajectories and predicted solutions to non-cooperative games under available hypotheses. In simulation, we consider three safety-critical autonomous driving scenarios of overtaking, lane-merging and intersection crossing, to demonstrate our planner's ability to capitalize on alternative intention hypotheses to generate safe trajectories in the presence of faulty transmissions in the communication network.

Index Terms—Path Planning for Multiple Mobile Robots or Agents, Planning under Uncertainty, Collision Avoidance.

I. INTRODUCTION

ROBOT control in interactive environments is very challenging due to the complexity of evaluating the impact of an agent's actions on the behaviour of others. Predict-then-plan strategies have substantially been addressed in the literature [1], [2]; trajectories of other agents are predicted first, then used as constraints in a single-agent planning scheme. However this decoupling ignores the inherently interactive nature of the problem. Indeed, in order to capture the reactive nature of agents in a scene, a robot must be able to simultaneously predict the trajectories of other agents while planning its own trajectory. Differential game theory provides a suitable framework for expressing these types of multi-agent planning problems without requiring *a priori* predictive assumptions.

The main challenge of decentralized implementation of game-theoretic motion planners is that common knowledge of the dynamics, constraints and objective functions of all partaking players is required. One can argue that dynamics and constraints could be assumed or inferred in structured environments e.g. in driving scenarios. However, for a player
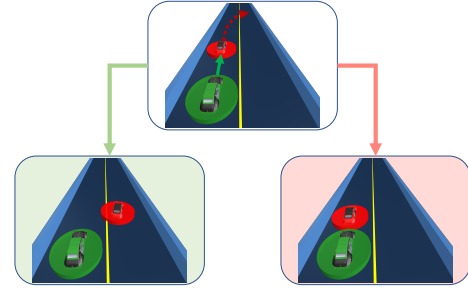
Fig. 1: Overtaking on a highway. As the red car signals to the right, the green car anticipates it making way and accelerates. If the red car's signalling is reliable and it does move over, the green car would have capitalized on this communicated intention to reach its objective faster (left scenario). On the other hand, if the red car does not switch lanes, the green car runs the risk of collision if it is unable to react in time (right scenario).

to assign an objective function to each of the other agents, they must have a representation of their intentions. We propose to use vehicle-to-vehicle (V2V) communication for vehicles to share their intentions with one another.

In the game-theoretic framework, the communication of intention is best formalized as the sharing of one's cost function, which directly intervenes in computing the Nash equilibrium trajectory solution. However, using communication network between agents raises questions about the security and robustness of the communication protocol to breaches and faults in safety-critical applications. A simple and very frequent real-life example is that of overtaking, where knowledge of the other vehicle's intention allows for a more efficient maneuver as shown in Fig. 1. In this example, we present a case where faulty communication could lead to a collision.

In this paper, we investigate the use of V2V communication as a solution to the safety shortcomings arising from cost function mismatch in Model Predictive Control (MPC) implementations of game-theoretic motion planners. We specify the requirements such a solution would need to satisfy, namely robustness to communication faults and adaptability throughout the unfolding of an evolving multi-agent scene. In this work, we assume the vehicles are connected using V2V communication network. Also, we assume a set of possible intentions (hypotheses) for each vehicle is available to be used in real time. Our proposed planner updates hypothesis likelihoods, allowing an agent to reassess the reliability of communicated intentions coming from each player in the scene. With a hypothesis likelihood assigned to players' intentions, the agent adapts its version of a dynamic game at each replanning interval. Our main contributions are,

1) An online Bayesian inference approach to estimate the relative likelihoods of available hypotheses based on past observations in the context of dynamic games with communicated cost functions,
2) A communication-based game-theoretic receding horizon motion planner is designed to be resilient to faulty transmissions and safer than naive MPC implementations of dynamic game motion planners,
3) Simulations on three autonomous driving scenarios: overtaking, lane-merging and intersection crossing, empirically showcasing the superiority of our approach in planning safe trajectories and avoiding crashes.

II. RELATED WORK

Dynamic Games. Differential game theory provides a framework to model multi-agent interactions without requiring a priori assumption about predicting other agents' behaviors [3]. Differential games require solving coupled Hamilton-Jacobi equations which are computationally intractable for high-dimensional spaces. To make the problem tractable, iterative linear-quadratic (iLQ) games are proposed in [4],[5]. The iterative linear quadratic games provide feedback strategies to form a feedback Nash equilibrium of the game. These approaches are computationally efficient, but since the feedback policies are fixed, they cannot adapt to any change in the behavior of the other agents. In contrast, our proposed approach allows adaptability to any change in the game structure and provides open-loop Nash equilibrium, instead.

Game-theoretic planning A game-theoretic setting allows to model both adversarial and cooperative agents. In fact, scenarios such as autonomous driving, where robots need to interact with other intelligent agents, are fundamentally game-theoretic [6], [7], [8]. Formalizing such interactions as a game, robots can weigh the impact of their decisions on the actions of other agents [9], even in competitive settings such as drone or car racing [10], [11]. Most game-theoretic works do not assume availability of a communication network among the agents and model the interactions as a game formulation. In [12], [13], [14] data-driven methods are presented to infer objective functions of other agents online in game theoretic planning without communication. On the other hand distributed optimization approaches assume the existence of a communication network and solve the optimization based on the received communicated predictions [15], [16]. They devise a communication and planning protocol in which all the agents share their predicted planned trajectories. One drawback of the former approach is that in an uncertain and dynamic environment such as autonomous driving setting, solving a game with a fixed structure may not be robust with respect to adversarial agents' behavior.

Also, distributed optimization approaches that completely rely on the communicated trajectories and make use of predictions in the planning are susceptible to adversarial cyber-attacks. In contrast, our proposed approach considers a combination of the two, in which we consider solving a game while a communication network is assumed among the agents and they make use of communicated information to update their reliability towards other agents. Within the receding-horizon game-theoretic framework the agents update the game cost function according to the communicated information about other agents' intentions. Incorporating reliability into the game formulation leads to quick adaptation and promotes resilience against failures.

Communication-based planning and control. The emergence of connected and automated vehicles [17], [18] has the potential to drastically improve mobility, energy efficiency and safety, especially in traffic bottlenecks (such as merging, intersection, etc.) Optimal control problem are used in some of these approaches [19], while MPC techniques are employed as an alternative [20], mainly to take additional constraints and disturbances into account. All the above mentioned works assume that information shared through communication network is reliable. While in this work, we consider communication-based control in the presence of faulty transmissions on the communication network.

III. PROBLEM STATEMENT

We consider a sequence of Generalized Nash Equilibrium Problems (GNEPs) involving N players $i \in \{1, \ldots, N\}$ over a horizon of H time steps. An agent i's state at time step index t is denoted $\mathbf{x}_t^i \in \mathbb{R}^{n^i}$ and control input $\mathbf{u}_t^i \in \mathbb{R}^{m^i}$, with dimensions of agent i's state and control n^i and m^i. Let $\mathbf{x}_t = [\mathbf{x}_t^{1,\top}, \ldots, \mathbf{x}_t^{N,\top}]^\top \in \mathbb{R}^n$ denote the joint state and $\mathbf{u}_t = [\mathbf{u}_t^{1,\top}, \ldots, \mathbf{u}_t^{N,\top}]^\top \in \mathbb{R}^m$ denote the joint control of all agents at time t, with joint dimensions $n = \sum_i n^i$ and $m = \sum_i m^i$. We define player i's policy as $\pi^i = [\mathbf{u}_1^{i,\top}, \ldots, \mathbf{u}_{H-1}^{i,\top}]^\top \in \mathbb{R}^{\tilde{m}^i}$ where $\tilde{m}^i = m^i(H-1)$ denotes the dimension of the entire trajectory of agent i's control inputs. The notation $\neg i$ indicates all agents except i, for instance $\pi^{\neg i}$ represents the vector of the agents' policies except that of i. Also, let $X = [\mathbf{x}_2^\top, \ldots, \mathbf{x}_H^\top]^\top \in \mathbb{R}^{\tilde{n}}$, with $\tilde{n} = n(H-1)$, denote the trajectory of joint state variables resulting from the application of the joint control inputs to the dynamical system defined by $f : \mathbb{R}^n \times \mathbb{R}^m \to \mathbb{R}^n$ such that,

$$\mathbf{x}_{t+1} = f(\mathbf{x}_t, \mathbf{u}_t) \tag{1}$$

Over the whole trajectory we can express the above kinodynamic constraints with $\tilde{n}$ equality constraints,

$$D(X, \pi^1, \ldots, \pi^N) = D(X, \pi) = 0 \in \mathbb{R}^{\tilde{n}} \tag{2}$$

The cost function of each player i depends on its policy π^i as well as on the joint state trajectory X, which is common to all players, such that $\forall i \in \{1, \ldots, N\}$,

$$J^i(X, \pi^i) = c_H^i(\mathbf{x}_H) + \sum_{t=1}^{H-1} c_t^i(\mathbf{x}_t, \mathbf{u}_t^i) \tag{3}$$

Notice that as player i minimizes J^i with respect to X and π^i, the selection of X is constrained by the other players' strategies $\pi^{\neg i}$ and the dynamics of the joint system via (2). In addition, the strategy π^i could be required to satisfy (safety) constraints that depend on the joint state trajectory X as well as on the other players strategies $\pi^{\neg i}$. This can be expressed with a set of g inequality constraints,

$$C(X, \pi) \leq 0 \in \mathbb{R}^g \tag{4}$$

where $C : \mathbb{R}^{\tilde{n}} \times \mathbb{R}^{\tilde{m}} \to \mathbb{R}^g$. The GNEP we form is the problem of minimizing (3) for all players $i \in \{1, \ldots, N\}$ with respect to (2) and (4). More specifically,

$$\min_{X, \pi^i} J^i(X, \pi^i) \qquad \forall i \in \{1, \ldots, N\}$$
$$\text{subject to} \quad D(X, \pi) = 0 \qquad (5)$$
$$C(X, \pi) \leq 0$$

The solution to such a dynamic game is a generalized Nash equilibrium, i.e. a policy π such that, $\forall i \in \{1, \ldots, N\}$, π^i is a solution to (5) with the other players' policies given by $\pi^{\neg i}$ that is also solved by (5) for all $\neg i$. As a consequence, at a Nash equilibrium point solution, no player can improve their strategy by unilaterally modifying their policy.

MPC implementation with cost updates

Cost structures are fixed during the resolution of a dynamic game, which can lead to dangerous trajectories when costs used differ from the true expressions. Hence, intelligent agents should be able to update their interpretation of others' intentions. In our game-theoretic framework, this is equivalent to enabling objective function estimate updates at relevant frequencies. Proposed methods to achieve this include inverse optimal control algorithms and Inverse Reinforcement Learning (IRL). To the best of our knowledge, this problem has not yet been tackled for intention-communicating agents with possible transmission faults corrupting the communication channel. We design a communication-based game-theoretic receding horizon motion planner to update the cost representations agents have of others by capitalizing on truthful communication and absorbing misleading transmissions to avoid dangerous trajectories.

IV. HYPOTHESIS ADAPTIVE MOTION PLANNER

Our solution relies on updating the perceived costs by comparing executed trajectories to those obtained by solving the set of games corresponding to various intention hypotheses. The approach is similar in principle to type-based games [21], where the type of an agent is their intention, which, in our case coincides with their cost function, thus constituting a complete specification of the agent's behaviour. This approach is based upon the observation that the set of broadly defined possible intentions in many multi-robot motion planning is small, and that solutions that are close most likely arise from cost formulations that are also close. The latter assumption need not hold, not only for Nash equilibria but also for individual agents planning in the presence of obstacles as the mapping between cost formulations and corresponding trajectory solutions will generally not be smooth. For instance, as we increase the desired velocity of an ego vehicle initially merging behind an ado vehicle, a threshold value will be reached, beyond which the ego accelerates to merge in front of the ado. Values infinitesimally close on either side of this threshold thus lead to disparate solutions. These discontinuity points are however very sparse and such cases are extremely unlikely to randomly occur in simulation, where behaviour consistent with our assumption is observed. The non-uniqueness of Nash

equilibria also means that even with correct costs, potentially large trajectory deviations can arise. However, Peters et al. [22] show that it is possible to ensure agents agree on a Nash equilibrium.

We denote agent i's true cost function, used for their own planning in (5), as c_v^i. Agent i broadcasts a communicated cost denoted c_c^i to all other agents in its vicinity. Communication need not be truthful, i.e. we do not constrain $c_v^i = c_c^i$. We also have access in a database to a set of ξ possible hypotheses for other agents intentions, notably for agent i, cost functions $c_{h,l}^i$ for $l \in \{1, 2, \ldots, \xi\}$. We denote c_h^i the vector of hypothetical cost functions agent i has about the scene such that $\mathbf{c}_h^i = [c_{h,1}^i, \ldots, c_{h,\xi}^i]$. Agent i solves a recursive game and must estimate $\xi + 1$ probability parameters representing the relative likelihood of each of the hypotheses, λ_l with $l \in \{0, \ldots, \xi\}$ and such that $\sum_{l=0}^{\xi} \lambda_l = 1$ and $\lambda_l \geq 0$, then uses the estimated likelihoods to assign a cost for agent j denoted c_j^i. The zeroth index corresponds to the communicated intention hypothesis such that probability λ_0 is as associated with cost c_c^j. The ego vehicle solves a non-cooperative game with its own cost c_v and the assigned cost function c_j (we omit the superscript i for clarity) for an ado agent j according to,

$$k_{\max} = \text{argmax}([\lambda_0, \ldots, \lambda_\xi]) \qquad (6)$$

$$c_j = \begin{cases} c_c^j & , \quad \text{if } k_{\max} = 0 \\ c_{h, k_{\max}} & , \quad \text{otherwise} \end{cases} \qquad (7)$$

The design choice of planning with the highest likely hypothesis is motivated by the discrete nature of possible cost functions in most driving scenarios (switching lanes, turning left or right etc.). In games with continuous cost structure (desired speed, aggressiveness, comfort metrics), we propose using the weighted sum of the intention hypotheses to determine the cost assigned to agent j

$$c_j = \lambda_0 c_c^j + \sum_{l=1}^{\xi} \lambda_l c_{h,l} \qquad (8)$$

A. Online Bayesian inference for hypothesis estimation

We use online discrete Bayesian filtering to update the distribution weights. The belief vector $\mathbf{\Lambda}$ is defined as $\mathbf{\Lambda} = [\lambda_0, \lambda_1, \ldots \lambda_\xi]^T$. The transition matrix is defined as $\mathbf{T} = [T_{kl}]_{k,l=0}^{\xi}$, where each entry of the matrix is $T_{kl} = p(l_{t+1} = k | l_t = l)$. The filter predict step is,

$$\mathbf{\Lambda}_{t+1|t} = \mathbf{T} \mathbf{\Lambda}_{t|t} \qquad (9)$$

where the subscript $t|t'$ denotes the estimate of $\mathbf{\Lambda}$ at time t given the observations up to and including time t'. To make minimal assumption on how the intention dynamics evolve, we take $\mathbf{T}$ to be the identity matrix. The measurement matrix $\mathbf{M}$ is defined as, $\mathbf{M} = \text{diag}[p(\mathbf{y}_{t-s:t} | I_t = 0), \ldots, p(\mathbf{y}_{t-s:t} | l_t = \xi)]$, where the measurement likelihood for diagonal elements of the matrix $\mathbf{M}$ are computed based on the steps outlined in Algorithm 1. The filtering equations are formulated over s state measurements and the filter update step is,

$$\mathbf{\Lambda}_{t+1|t+1} = \frac{\mathbf{M} \mathbf{\Lambda}_{t+1|t}}{1^T \mathbf{M} \mathbf{\Lambda}_{t+1|t}} \qquad (10)$$

In Algorithm 1, the inputs are measurements $\mathbf{y}_{t-s:t}$ and belief vector $\mathbf{\Lambda}_{t-s}$ over the past s time steps. The output is the measurement likelihood $p(\hat{\mathbf{x}}_{t-s:t} = \mathbf{y}_{t-s:t}|\mathbf{\Lambda}_{t-s})$, where $\hat{\mathbf{x}}$ denotes the open-loop Nash solution of the dynamic game (5). The game is solved over H horizon from $\mathbf{y}_{t-s}$ with $\mathbf{\Lambda}_{t-s}$ and the control sequence $\mathbf{u}_{t-s:t+H}$ is computed. Then the dynamics is propagated forward up to time t and the state trajectory $\hat{\mathbf{x}}_{t-s:t}$ is computed. Finally the measurement likelihood $p(\hat{\mathbf{x}}_{t-s:t} = \mathbf{y}_{t-s:t}|\mathbf{\Lambda}_{t-s})$ is computed.

Algorithm 1 Measurement likelihood for discrete Bayesian filter

---

1: **Input:** $\mathbf{y}_{t-s:t}, \mathbf{\Lambda}_{t-s}$
2: **Output:** $p(\hat{\mathbf{x}}_{t-s:t} = \mathbf{y}_{t-s:t}|\mathbf{\Lambda}_{t-s})$
3: **Steps:** $\mathbf{u}_{t-s:t+H} \leftarrow$ solve game (5) from $\mathbf{y}_{t-s}$ with $\mathbf{\Lambda}_{t-s}$ over H horizon
4: $\hat{\mathbf{x}}_{t-s:t} \leftarrow$ forward propagate (1) up to time t
5: $p(\hat{\mathbf{x}}_{t-s:t} = \mathbf{y}_{t-s:t}|\mathbf{\Lambda}_{t-s}) \leftarrow$ likelihood with $p(\mathbf{x}_{t-s:t}|\mathbf{\Lambda}_{t-s}) \propto p(\mathbf{x}_{t-s:t}|\hat{\mathbf{x}}_{t-s:t}(\mathbf{u}_{t-s:t}(\mathbf{\Lambda}_{t-s})))$

In our setup, at time step $t - s$ agent i solves his current perceived game determined by $\mathbf{\Lambda}_t$ according to (6)-(7) with horizon H and executes the trajectory obtained for $s < H$ time steps. At step t, agent i observes the executed joint state trajectories of all agents between $t - s$ and t which we denote $\hat{X}_{t-s:t} = [\hat{\mathbf{x}}_{t-s}^{\top}, \dots, \hat{\mathbf{x}}_t^{\top}]^{\top}$. Similarly, for $l \in \{0, 1, \dots, \xi\}$ agent i can obtain hypothetical trajectories $X_{t-s:t}^l = [\mathbf{x}_{t-s}^{l,\top}, \dots, \mathbf{x}_t^{l,\top}]^{\top}$ by solving each of the available versions of the game, with the convention that $l = 0$ represents the communicated game. We can thus compute, for hypotheses $l \in \{0, 1, \dots, \xi\}$, a disparity score with the observed executed joint trajectory d^l, which we take to be the sum along all points of the trajectory if the $l1$-norm of the element-wise relative state error, such that,

$$d^l = \sum_{k=t-s}^{t} \left\| (\mathbf{x}_k^l - \hat{\mathbf{x}}_k) \oslash \hat{\mathbf{x}}_k \right\|_1 \quad (11)$$

where the $\oslash$ symbol represents the Hadamard element-wise division. The choice of this formulations allows for all elements of the observed state (positions, translation velocities, angular rates. etc.) to have the same weight in the norm computation. We shed light on the necessity of taking numerical precautions to avoid divisions by zero in (11). We also ensure all computed error scores are in practice strictly positive by adding a small offset to (11). The hypothesis relative likelihoods λ_l for $l \in \{0, 1, \dots, \xi\}$ presented in Section IV-A must rank in the inverse order of that of the disparity scores as closer trajectories are assumed to arise from games with more accurate cost function representations. We obtain our current estimate of the relative likelihoods denoted $\tilde{\mathbf{\Lambda}}_t = [\tilde{\lambda}_0, \dots, \tilde{\lambda}_\xi]$ by requiring likelihoods to be inversely proportional to disparity scores and thus to satisfy the following conditions,

$$\tilde{\lambda}_l = \frac{d^{l'}}{d^l} \tilde{\lambda}_{l'}, \quad \forall l, l' \in \{0, 1, \dots, \xi\}, \, l \neq l' \quad (12)$$

$$\sum_{l=0}^{\xi} \tilde{\lambda}_l = 1 \quad (13)$$

$$\tilde{\lambda}_l \geq 0, \quad \forall l \in \{0, 1, \dots, \xi\} \quad (14)$$

Note that the $\binom{\xi+1}{2}$ conditions ($\xi + 1$ choose 2) in (12) are highly redundant and can be replaced by ξ conditions:

$$\tilde{\lambda}_l = \frac{d^0}{d^l} \tilde{\lambda}_0, \quad \forall l \in \{1, \dots, \xi\} \quad (15)$$

Also the strict positivity of d^l for $l \in \{0, 1, \dots, \xi\}$ combined with condition (13) ensures that any solution to (15) and (13) necessarily satisfies (14). Thus, computing the current estimate of the relative likelihoods comes down to solving the simple linear system

$$\mathbf{A}_t \tilde{\mathbf{\Lambda}}_t^{\top} = \mathbf{B} \quad (16)$$

with $\mathbf{A}_t \in \mathbb{R}^{(\xi+1) \times (\xi+1)}$ and $\mathbf{B} \in \mathbb{R}^{\xi+1}$ such that,

$$\mathbf{A}_t = \begin{bmatrix} 1 & 1 & 1 & \cdots & 1 \\ 1 & -\frac{d^1}{d^0} & 0 & \cdots & 0 \\ 1 & 0 & \ddots & & \vdots \\ \vdots & \vdots & & \ddots & 0 \\ 1 & 0 & \cdots & 0 & -\frac{d^\xi}{d^0} \end{bmatrix}, \quad \mathbf{B} = \begin{bmatrix} 1 \\ 0 \\ \vdots \\ \vdots \\ 0. \end{bmatrix} \quad (17)$$

The likelihood estimation scheme is based on the online filtering approach presented in section IV-A. For the filter predict step (9), the transition probability matrix is assumed as identity matrix $\mathbf{T} = \mathbf{I}_{\xi+1}$ to make minimal assumption on the transition dynamics between different hypothesis. Therefore $\mathbf{\Lambda}_{t+1|t} = \mathbf{\Lambda}_{t|t}$. Equation (16) is equivalent to the filter update step (10). While the measurement likelihood $p(\hat{\mathbf{x}}_{t-s:t} = \mathbf{y}_{t-s:t}|\mathbf{\Lambda}_{t-s})$ discussed in algorithm 1 is computed using equation (11). We finally obtain the new value of $\mathbf{\Lambda}_t$ used to define the next perceived game to solve via the fusion of the previous value $\mathbf{\Lambda}_{t-s}$ with the new estimate $\tilde{\mathbf{\Lambda}}_t$ according to a fixed update rate γ, such that,

$$\mathbf{\Lambda}_t = (1 - \gamma)\mathbf{\Lambda}_{t-s} + \gamma \tilde{\mathbf{\Lambda}}_t \quad (18)$$

The update rate γ is the filter's only hyperparameter. It determines how fast we update our likelihood vector according to previous estimates. Smaller γ values (slower) can be useful in scenarios with high replanning frequencies, where shorter observed trajectories contain less useful signal, and a gradual convergence of $\mathbf{\Lambda}$ over a larger sample is desired. Larger γ values favor quick adaptation to new observations contradicting the current active hypothesis.

B. Proposed Motion Planner

Our solution consists of an MPC implementation of the open-loop dynamic game solver over a time horizon H, with the Nash equilibrium policy subsequently executed for H_N time steps before updating the perceived game costs and repeating. Agent i, with cost function c^i, initial joint state of the scene $\mathbf{x}_0$, initial hypothesis likelihood vector $\tilde{\mathbf{\Lambda}}_0$, and hypothesis cost models for other agents c_h^i, thus computes its motion according to our decentralized on-board planner outlined in Algorithm 2.

Algorithm 2 Planner for robot i

1: **procedure** PLANNER$(\mathbf{x}_0, \mathbf{\Lambda}_0, c^i, c_h^i, H, H_N, \gamma)$
2:     $\mathbf{x} \leftarrow \mathbf{x}_0, \quad \mathbf{\Lambda} \leftarrow \mathbf{\Lambda}_0$
3:     **while** *interacting* **do**
4:         COMMUNICATE(c^i)
5:         $c_c^{-i} \leftarrow$ RECEIVECOMMUNICATEDCOSTS
6:         $c \leftarrow$ FORMPERCEIVEDGAME$(\mathbf{\Lambda}, c_c^{-i}, c_h^i)$ (6)-(7)
7:         $\pi \leftarrow$ SOLVEDYNAMICGAME$(\mathbf{x}, c, H)$ (5)
8:         $\mathbf{x}^i \leftarrow$ EXECUTE$(\mathbf{x}^i, \pi^i, H_N)$
9:         $\mathbf{x}^{-i} \leftarrow$ OBSERVESTATE(H_N)
10:       $\mathbf{\Lambda} \leftarrow$ UPDATELIKELIHOOD$(\pi, \mathbf{x}, \gamma)$ (16)-(18)
11:     **end while**
12: **end procedure**

The hypothesis likelihood update compute time is largely negligible relative to the dynamic game solver. Hence, the planning time scales linearly with the number of hypotheses as compared to using ALGAMES to plan a single version of the dynamic game. Indeed, for a single version of a 3 agent game, a 20 step look-ahead and a 5 step executed trajectory, the compute costs required for solving the game and updating the hypotheses likelihood vector are shown in Table I.

TABLE I: Orders of magnitude of the run time, number of allocations and disk space required to solve the dynamic game and to update the hypothesis likelihood vector.

Task	Run time (s)	# alloc.	Space (MB)
SOLVEDYNAMICGAME	10^{-1}	10^5	100
UPDATELIKELIHOOD	10^{-4}	10^3	0.1

Also, ALGAMES solver has a $O(HN^3)$ complexity, i.e. linear in H the game horizon and cubic in N the number of agents. Our planner solves Ξ such games at each replanning time step, where Ξ is the number of hypotheses available to the planning agent. Thus, overall, our solution scales like $O(H\Xi N^3)$, and experimental compute times are given in Table II.

TABLE II: Ego vehicle run times (average for 5 runs, in ms) for a fixed horizon H = 20 for the intersection scenario with 2,3 and 4 agents with an increasing number of hypotheses from 1 to 3.

	$N = 2$	$N = 3$	$N = 4$
$\Xi = 1$	3	142	776
$\Xi = 2$	5	272	1533
$\Xi = 3$	8	403	2268

V. SIMULATIONS

We test our proposed solution on three autonomous driving scenarios inspired by real-life situations with increasing complexity. In our simulations we introduce confusion in the communication scheme to show the adaptation ability of our planner in the case of cost mismatch. We look at a two vehicle overtaking maneuver, a three vehicle lane merging scenario and a four vehicle intersection negotiation. In all the simulations, all agents plan their trajectories using our proposed planner.

A. Simulation setup

The dynamic game solver we use is ALGAMES [23], which handles trajectory optimization problems with multiple actors and general nonlinear state and input constraints. The vehicles obey nonlinear unicycle dynamics. The state of a vehicle $\mathbf{x}_t^i$ comprises of its 2D position (*px* horizontal, *py* vertical), its heading angle and scalar velocity. The control input $\mathbf{u}_t^i$ comprises of the angular velocity and scalar acceleration.

1) Constraints: The dynamics constraints at time t consist in following the system dynamics given by (1), with f being unicycle dynamics in all our driving simulations. We also enforce collision-avoidance constraints on the trajectories, by modelling collision zones of the vehicles by circles or radius r, such that, at any time step t, $\left\| \mathbf{p}_t^i - \mathbf{p}_t^j \right\|_2^2 \geq r^2, \quad \forall i, j \in \{1, \ldots, N\}$. In addition, we require the vehicles to remain on the road, by constraining the distance between each vehicle and the closest point $\mathbf{q}$ on each boundary b to remain larger than the collision radius r, $\left\| \mathbf{p}_t^i - \mathbf{q}_b \right\|_2^2 \geq r^2, \quad \forall b, \forall i \in \{1, \ldots, N\}$ where $\mathbf{p}_t^i = [px_t^i, py_t^i]$ contains the plane coordinates of the agent i at time t extracted from the complete state vector $\mathbf{x}_t^i$. Thus, the autonomous driving problem is formalized via non-convex and non-linear coupled constraints.

2) Cost function: The cost structure considered is quadratic, penalizing the distance to the desired final state $\mathbf{x}_f$ and the use of controls, $J^i(X, \pi^i) = \sum_{t-1}^{H-1} \frac{1}{2}(\mathbf{x}_t - \mathbf{x}_f)^\top \mathbf{Q}(\mathbf{x}_t - \mathbf{x}_f) + \frac{1}{2}\mathbf{u}_t^\top \mathbf{R}\mathbf{u}_t + \frac{1}{2}(\mathbf{x}_H - \mathbf{x}_f)^\top \mathbf{Q}_f(\mathbf{x}_H - \mathbf{x}_f)$, where $\mathbf{Q}$, $\mathbf{R}$ and $\mathbf{Q}_f$ represent state, input and final state penalization weight matrices, respectively. This cost function depends only on the decision variables of vehicle i, as players' behaviors are only coupled through collision constraints. Thus, although knowledge of other agents' intentions does not intervene in the individual cost agent i is optimizing for, it does however determine the trajectories of others, and subsequently the collision constraints agent i has to satisfy.

B. Advantage of V2V communication

We show through a simple takeover maneuver experiment that intention sharing between agents can improve the quality of the planned solutions. Vehicle v_1 wishes to maintain high velocity on the left lane along a two-lane highway. Vehicle v_2 is in front of it initially and is driving at a slower speed. v_2 wishes to move to the right lane. We consider two scenarios: first with v_1 agnostic to v_2's intention, and second with v_2's true cost function available to v_1. In the first case, since no information is available to v_1, we let v_1 suppose that v_2's desired state is on the left lane at the same speed. The trajectories in both cases are presented in Fig. 3. In this figure and all subsequent figures depicting executed trajectories, the marker size increases with respect to time.

We clearly notice that access to v_2's intentions allows v_1 to satisfy its objective of maintaining high speed on the left. On the other hand, the absence of such information and of the ability to update the cost based on observations deteriorates the reward collected as v_1 gets stuck behind v_2, expecting the latter to drift back to the left lane.
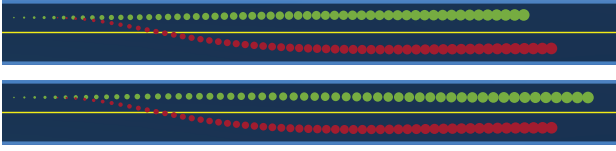
Fig. 3: No communication (top) and true cost communication (bottom). v_1 and v_2 executed trajectories in green and red respectively.

C. Unique true alternative

We first validate the soundness of our approach by considering cases with one agent sending faulty communication. The other agents use our planner with one alternative hypothesis, which we here assume to be the true cost function of the faulty agent. Furthermore, we assume that all players initially trust each other (at $t = 0$ we initialise $\lambda_0 = 1, \lambda_1 = 0$). We simulate three real-life driving situations: a takeover maneuver, a lane merging, and an intersection negotiation.

1) Takeover: Vehicle v_1 wishes to maintain high velocity on the left lane along a two-lane highway. In front of it on the same lane is a slower vehicle, v_2, which misleadingly expresses its intention to make way (e.g. by signalling to the right), when its true intention is to stay on the left lane.

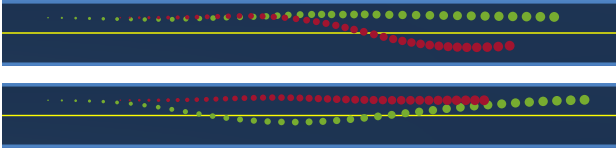The obtained trajectories with and without the use of our planner are presented in Fig. 4.

Fig. 4: Communication only planner solution (top) and adaptive hypothesis likelihood planner with $\gamma = 0.6$ solution (bottom). v_1 and v_2 executed trajectories in green and red respectively. The true desired states and quadratic state penalties are $x_f^1 = [6.0, 0.1, 0.0, 0.9]$, $Q^1 = Diag([0.0, 1.0, 0.1, 10.0])$ and $x_f^2 = [6.0, 0.1, 0.0, 0.6]$, $Q^2 = Diag([0.0, 20.0, 0.1, 1.0])$. v_2's communicated desired state is $x_{com}^2 = [6.0, -0.1, 0.0, 0.6]$.

Relying on communicated cost functions only, the overtaking vehicle v_1 persists in attempting to pass by v_2 from the left as at each episode the Nash equilibrium with the fixed communicated cost from v_2 would yield an optimal policy in which v_2 moves out of v_1's way. This leads to an aggressive maneuver where v_1 repeatedly bumps v_2 pushing it to the right lane for v_1 to squeeze through. Using our planner, v_1 observes v_2's reluctance to make way over the first few iterations, and quickly assigns the alternative cost function to v_2, leading to a smooth overtake from the right.

2) Lane-merging: Next we consider a lane-merging maneuver involving three vehicles on a two-lane highway. Let v_1 be the source of the fault, communicating its willingness to make way to the merging vehicle v_2, when in truth it wishes to maintain driving along the right lane without reducing its speed. Driving along the left lane, v_3's objective is to maintain high velocity.

Fig. 5 shows the trajectory obtained using communicated intentions only, and that with v_2 and v_3 using our planner. The first trajectory depicts an aggressive and unsafe merging scenario. Indeed, at each planning step, v_2 persists in assuming
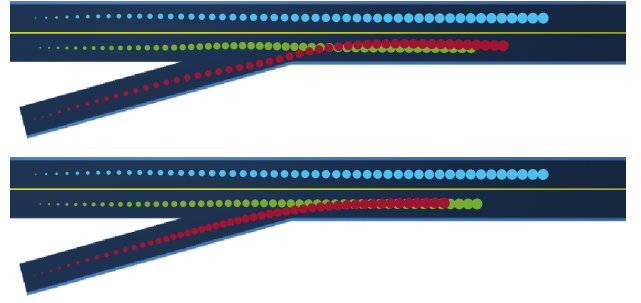

Fig. 5: Communication only planner solution (top) and adaptive hypothesis likelihood planner with $\gamma = 0.6$ (bottom) cases : v_1, v_2 and v_3 executed trajectories in green, red, and blue respectively. The true desired states and quadratic state penalties are $x_f^1 = [2.5, -0.1, 0.0, 0.6]$, $Q^1 = Diag([0.0, 20.0, 0.1, 20.0])$, $x_f^2 = [2.5, -0.1, 0.0, 0.6]$, $Q^2 = Diag([0.0, 1.0, 0.1, 1.0])$ and $x_f^3 = [3.0, 0.1, 0.0, 0.7]$, $Q^3 = Diag([0.0, 1.0, 0.1, 1.0])$. v_1's communicated cost matrix is $Q_{com}^1 = Diag([0.0, 1.0, 0.1, 1.0])$.

v_1 will either slow down or switch to the left lane in order to accommodate it. Hence, v_2 continues to execute the merge in front of v_1, passing dangerously close to both v_1 and the road boundary. Using our adaptive planner, v_2, notices v_1's reluctance to make way over a couple of planning episodes, assigns larger likelihood to the alternative scenario and subsequently plans to allow v_1 to pass first before merging. In this case, v_2 avoids finding itself in the hazardous situation of being tailgated by v_1.

3) 4-way intersection: The last example we consider involves four vehicles negotiating an intersection crossing. Three vehicles, v_1, v_2 and v_3 intend to drive straight across (west to east, east to west and north to south respectively). Vehicle v_4 wants to execute a left turn (south to west). We assume v_4 misleads the other agents about its aggressiveness. Trajectories planned with the faulty intention as well as those obtained with agents v_1, v_2 and v_3 using our adaptive planner are presented in Fig. 6.
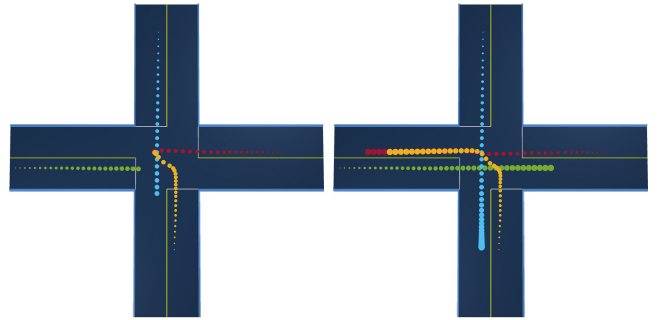

Fig. 6: Communication only planner solution (left) and adaptive hypothesis likelihood planner with $\gamma = 0.4$ (right) cases : v_1, v_2, v_3 and v_4 executed trajectories in green, red, blue and yellow respectively. The true desired states and quadratic state penalties are $x_f^1 = [1.2, -0.1, 0.0, 0.7]$, $Q^1 = Diag([0.0, 1.0, 0.1, 1.0])$, $x_f^2 = [-1.2, 0.1, -\pi, 0.7]$, $Q^2 = Diag([0.0, 1.0, 0.1, 1.0])$, $x_f^3 = [-0.1, -1.2, -\pi/2, 0.7]$, $Q^3 = Diag([0.0, 1.0, 0.1, 1.0])$ and $x_f^4 = [-1.2, 0.1, \pi, 0.7]$, $Q^4 = Diag([0.0, 1.0, 0.1, 20.0])$. v_4's communicated cost matrix is $Q_{com}^4 = Diag([0.0, 1.0, 0.1, 1.0])$.

Without adapting the cost functions, v_2 does not hasten its passage as v_4 comes increasing close to it, as communication

from v_4 suggest it is willing to slow down to let v_2 through. Thus v_2 drives right in front of v_4 as it executes its turn, leading to a collision. In the adaptive case, v_2, noticing v_4's absence of braking, accelerates as it enters the intersection and avoids getting crashed into.

4) Update rate dependency: We study the effect of the update rate γ on evolution of the perceived game determined by Λ and present quantitative analysis of the minimum distance between each pair of vehicles with respect to this filter parameter. Closed-loop simulations of our proposed planner for all three scenarios are presented in Fig. 7.
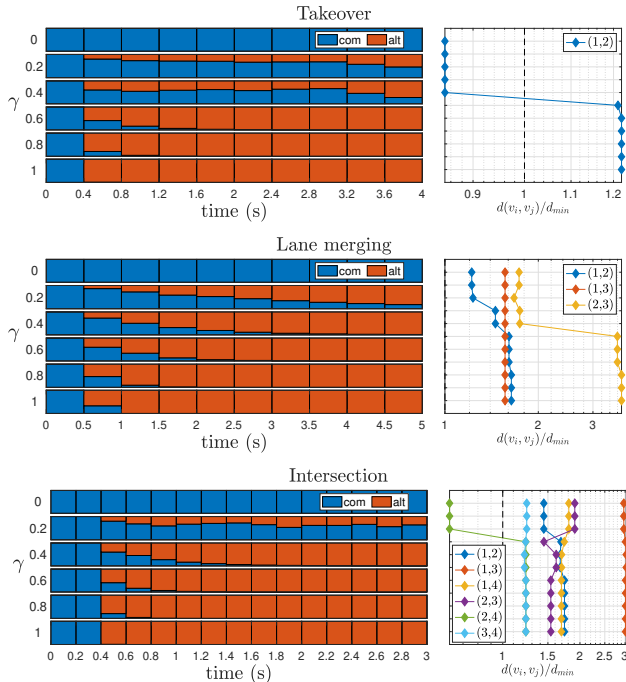

Fig. 7: Left: Evolution of the hypotheses relative likelihoods Λ throughout the simulation as a function of the filter update rate γ. The communicated cost's likelihood is depicted in blue, and the alternative hypothesis in red, the largest bar determining which version of the game is used for planning during the current episode. Right: Normalized distances between each pair of vehicles $d(v_i, v_j)/d_{min}$ (logarithmic scale) as a function of γ. The dashed line depicts the collision threshold d_{min}.

The main observation is that it is desirable for the hypothesis likelihood filter to converge before the start of the inter-vehicle interactions. Indeed, this not only allows agents to enjoy additional lead time to adapt their maneuvers to avoid collisions, but also to avoid interactions corrupting the intention signal used to update the filter. Overall, we notice that for large enough update rates, our planner allows agents to identify the correct intention of the faulty agent and use this information to replan safe trajectories.

D. Multiple inexact alternatives

We next test and validate our planner in the more complex, realistic setting of multiple hypotheses about other agents' intentions, which have no reason to necessarily include the exact true cost functions. We show that when the hypothesis set contains a cost function that is close enough to the true one, safer planning is achieved using our approach. Indeed,

we revisit the takeover example presented earlier, but now consider three alternative hypotheses to the communicated intention of v_2. The first is to maintain the left lane with a penalty on deviation from the lane center a quarter as small as the true one, the second is to maintain the left lane with a penalty 20 times smaller than the true value, and the third is to change lanes at a third of the initial speed. We show the evolution of Λ and of the minimum inter-vehicle distance as a function of γ in Fig. 8.
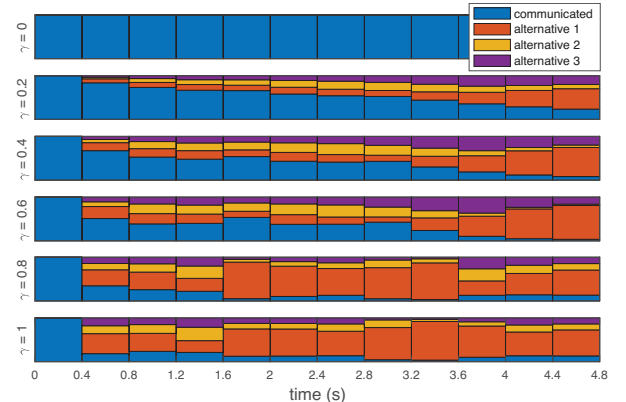

Fig. 8: Evolution of the hypotheses relative likelihoods Λ throughout the simulation as a function of the filter update rate γ. The communicated cost's likelihood is depicted in blue, and alternatives in red, yellow and purple. For $\gamma \geq 0.8$, collision is avoided.

According to the simulation results, our planner achieves collision-free trajectories in complex environments with broad and imprecise assumptions about other agents' costs utilizing the proposed online Bayesian filtering scheme which allows fast adaptation for the planner. The simulation results confirm that our solution can be used to ensure safety and improve performance in various driving scenarios.

E. Performance and robustness

We test the robustness of our approach to intention communication reliability, initial state conditions and process noise on the lane-merging scenario, comparing it to two planning baselines. We consider two available intention hypotheses for v_1 (I_1: v_1 wishes to stay on the right lane maintaining same speed, and I_2: v_1 wishes to accelerate with less emphasis on lane keeping), in addition to the communicated intention (I_c: v_1 is willing to yield the lane to the merging v_2). We assume all agents have access to the correct cost functions for v_2 and v_3. Our planner has access to all three hypotheses for planning. The first baseline is a communication-only planner using only I_c. The second uses our planner in a no-communication scheme, i.e. with only I_1 and I_2 as possible options. We uniformly randomize the initial positions of all three agents and introduce 20% uniform process noise on the executed control commands. We run 100 merging simulations with faulty communication (v_1's cost function is represented by I_1 and not I_c) and 100 runs with correct communication (v_1's cost is I_c as communicated). The results are presented in Table III.

As intended, our proposed planner generates safe trajectories whether the communicated intention is correct or not.

	Hyp.	d	Risky	Crash	J	acc_{max}
Faulty	$\{I_c\}$	1.40 ± 0.09	16%	0%	11.46	10.92
Faulty	$\{I_1,I_2\}$	1.56 ± 0.04	2%	0%	11.21	10.80
Faulty	$\{I_c,I_1,I_2\}$	1.55 ± 0.06	1%	0%	11.07	10.74
Correct	$\{I_c\}$	1.57 ± 0.01	0%	0%	11.09	10.66
Correct	$\{I_1,I_2\}$	1.54 ± 0.12	4%	2%	11.19	10.76
Correct	$\{I_c,I_1,I_2\}$	1.57 ± 0.01	0%	0%	11.07	10.66
Total	$\{I_c\}$	1.48 ± 0.10	8%	0%	11.28	10.79
Total	$\{I_1,I_2\}$	1.56 ± 0.09	3%	1%	11.20	10.78
Total	$\{I_c,I_1,I_2\}$	1.55 ± 0.05	0.5%	0%	11.07	10.70

TABLE III: Statistical performance of our planner versus communication and agnostic baselines ($n = 100$ maneuvers for each of Faulty/Correct communication scenarios).n d (mean $\pm$ std) represents the minimum normalized distance between each pair of vehicles, such that $d = \min\{d(v_i, v_j)/d_{min} \mid i, j \in \{1,2,3\}, i \neq j\}$, with d_{min} denoting the collision threshold. Risky maneuvers are accounted for when d is within 30% of d_{min}, and crashes when $d \leq d_{min}$. The cost of the maneuver for the merging agent, denoted J, is computed according to the quadratic cost. The maximum acceleration of the merging vehicle acc_{max} is used as a comfort score.

Indeed, over the 200 simulations with our adaptive planner, only one trajectory is considered risky and no crashes are recorded. Also as expected, our planner matches the perfect information game in terms of performance when communication is correct. We also notice the importance correct communication can have in insuring safety, as not having access to a close enough intention in the set of hypotheses can lead to dangerous trajectories and crashes. Indeed, when v_1 correctly communicates its intention I_c the adaptive planner using only $\{I_1, I_2\}$ exhibits a large variance in trajectories leading to a 2% crash rate. Our planner is capable of both absorbing situations with faulty intention communication, and leveraging correct information for Nash optimal trajectory planning, without compromising on performance and comfort.

VI. CONCLUSION

We have introduced a new receding horizon game-theoretic motion planner designed to operate in real-time in the framework of intention sharing multi-robot systems with potential communication faults. We demonstrate the robustness of our solution to such perturbations on complex autonomous driving scenarios, showcasing its ability to plan safe trajectories in situations from which naive implementations fail to recover. Future directions include integrating online intention identification into our framework in addition to incorporating state uncertainty.

REFERENCES

[1] G. Best and R. Fitch, "Bayesian intention inference for trajectory prediction with an unknown goal destination," in *2015 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2015, pp. 5817–5823.

[2] T. Bandyopadhyay, K. S. Won, E. Frazzoli, D. Hsu, W. S. Lee, and D. Rus, "Intention-aware motion planning," in *Algorithmic Foundations of Robotics X*, E. Frazzoli, T. Lozano-Perez, N. Roy, and D. Rus, Eds. Berlin, Heidelberg: Springer Berlin Heidelberg, 2013, pp. 475–491.

[3] A. W. Starr and Y. C. Ho, "Nonzero-sum differential games," *Journal of Optimization Theory and Applications*, vol. 3, no. 3, 1969.

[4] D. Fridovich-Keil, E. Ratner, L. Peters, A. D. Dragan, and C. J. Tomlin, "Efficient iterative linear-quadratic approximations for nonlinear multiplayer general-sum differential games," in *2020 IEEE International Conference on Robotics and Automation (ICRA)*, 2020, pp. 1475–1481.

[5] W. Schwarting, A. Pierson, S. Karaman, and D. Rus, "Stochastic dynamic games in belief space," *IEEE Transactions on Robotics*, vol. 37, no. 6, pp. 2157–2172, 2021.

[6] J. F. Fisac, E. Bronstein, E. Stefansson, D. Sadigh, S. S. Sastry, and A. D. Dragan, "Hierarchical game-theoretic planning for autonomous vehicles," *2019 International Conference on Robotics and Automation (ICRA)*, 2018.

[7] P. Trautman and A. Krause, "Unfreezing the robot: Navigation in dense, interacting crowds," in *2010 IEEE/RSJ International Conference on Intelligent Robots and Systems*, 2010, pp. 797–803.

[8] A. Dreves and M. Gerdts, "A generalized nash equilibrium approach for optimal control problems of autonomous cars," *Optimal Control Applications & Methods*, vol. 39, pp. 326–342, 2018.

[9] D. Sadigh, S. S. Sastry, S. A. Seshia, and A. D. Dragan, "Planning for autonomous cars that leverage effects on human actions," in *Robotics: Science and Systems*, 2016.

[10] R. Spica, E. Cristofalo, Z. Wang, E. Montijano, and M. Schwager, "A real-time game theoretic planner for autonomous two-player drone racing," *IEEE Transactions on Robotics*, vol. 36, pp. 1389–1403, 2018.

[11] M. Wang, Z. Wang, J. Talbot, J. C. Gerdes, and M. Schwager, "Game-theoretic planning for self-driving cars in multivehicle competitive scenarios," *IEEE Transactions on Robotics*, vol. 37, no. 4, pp. 1313–1325, 2021.

[12] L. Peters, D. Fridovich-Keil, V. Rubies-Royo, C. J. Tomlin, and C. Stachniss, "Inferring objectives in continuous dynamic games from noise-corrupted partial state observations," *arXiv preprint arXiv:2106.03611*, 2021.

[13] F. Laine, D. Fridovich-Keil, C.-Y. Chiu, and C. Tomlin, "Multi-hypothesis interactions in game-theoretic motion planning," in *2021 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 2021, pp. 8016–8023.

[14] S. Le Cleac'h, M. Schwager, and Z. Manchester, "Lucidgames: Online unscented inverse dynamic games for adaptive trajectory prediction and planning," *IEEE Robotics and Automation Letters*, vol. 6, no. 3, pp. 5485–5492, 2021.

[15] F. Rey, Z. Pan, A. Hauswirth, and J. Lygeros, "Fully decentralized admm for coordination and collision avoidance," in *2018 European Control Conference (ECC)*, 2018, pp. 825–830.

[16] L. Ferranti, R. R. Negenborn, T. Keviczky, and J. Alonso-Mora, "Coordination of multiple vessels via distributed nonlinear model predictive control," in *2018 European Control Conference (ECC)*, 2018, pp. 2523–2528.

[17] B. Schrank, B. Eisele, T. Lomax, and J. Bak. The 2015 urban mobility scorecard. Texas A&M Transportation Institute.

[18] Y. Bichiou and H. A. Rakha, "Developing an optimal intersection control system for automated connected vehicles," *IEEE Trans. on Intelligent Transportation Systems*, vol. 20, no. 5, pp. 1908–1916, 2019.

[19] W. Xiao and C. G. Cassandras, "Decentralized optimal merging control for connected and automated vehicles with safety constraint guarantees," *Automatica*, vol. 123.109333, 2021.

[20] I. A. Ntousakis, I. K. Nikolos, and M. Papageorgiou, "Optimal vehicle trajectory planning in the context of cooperative merging on highways," *Trans. Resear. part C*, vol. 71, pp. 464–488, 2016.

[21] "Autonomous agents modelling other agents: A comprehensive survey and open problems," *Artificial Intelligence*, vol. 258, pp. 66–95, 2018.

[22] L. Peters, D. Fridovich-Keil, C. J. Tomlin, and Z. N. Sunberg, "Inference-based strategy alignment for general-sum differential games," in *Proceedings of the 19th International Conference on Autonomous Agents and MultiAgent Systems*, ser. AAMAS '20. Richland, SC: International Foundation for Autonomous Agents and Multiagent Systems, 2020, p. 1037–1045.

[23] S. L. Cleac'h, M. Schwager, and Z. Manchester, "ALGAMES: A fast solver for constrained dynamic games," *CoRR*, vol. abs/1910.09713, 2019.