Path Planning Games

Yi Li and Yevgeniy Vorobeychik

Electrical Engineering and Computer Science, Vanderbilt University, Nashville, TN

Abstract. Path planning is a fundamental and extensively explored problem in robotic control. We present a novel *economic* perspective on path planning. Specifically, we investigate strategic interactions among path planning agents using a game theoretic path planning framework. Our focus is on economic tension between two important objectives: efficiency in the agents' achieving their goals, and safety in navigating towards these. We begin by developing a novel mathematical formulation for path planning that trades off these objectives, when behavior of other agents is fixed. We then use this formulation for approximating Nash equilibria in path planning games, as well as to develop a multi-agent cooperative path planning formulation. Through several case studies, we show that in a path planning game, safety is often significantly compromised compared to a cooperative solution.

Keywords: Multi-agent system · Path planning

1 Introduction

Path planning is a fundamental technical problem in autonomous robotic control. Decades of development have led to significant theoretical and algorithmic progress, with autonomous vehicles (including autonomous cars and UAVs) increasingly finding their way to urban roads and skies.

In much of the research on path planning, including mobile robot navigation [10, 1, 11], a fundamental task is to find a collision-free motion from a starting position to the goal position given a collection of known obstacles. Variations on this theme, such as dealing with stochastic and moving obstacles, have received recent attention with the emergence of numerous novel unmanned robotic systems and aerial vehicles [16, 2, 9].

As interactions among autonomous vehicles, be it on our roads or in the skies, becomes more routine, we can expect a certain amount of conflict to emerge, as the autonomous agents, designed in service of their individual goals, must occasionally find these goals dependent on other autonomous agents nearby. However, remarkably little research has been devoted to the question of what autonomous vehicle ecosystem would thereby emerge, when many autonomous agents attempt to achieve their individual goals, but must necessarily interact with one another in doing so.

To investigate the consequences of such *strategic* interactions among multiple path planners, we propose a study of *path planning games*. An important feature of such games is that a collection of self-interested path planners each trade off two objectives: efficiency, or speed with which their goals are achieved, and safety, or probability that they crash before reaching their goals. Moreover, they trade these off in individual, potentially diverse, ways. Consequently, in order to study path planning games we must take an *economic*, rather than a purely algorithmic, perspective on path planning.

To this end, we first develop a novel mathematical programming method for computing a single-agent path plan, accounting for these two objectives, given fixed dynamic behavior (i.e., path plans) of all other agents, as well stochastic disturbances in the environment. Next, we propose a simple iterative algorithm, best response dynamics, for approximately computing Nash equilibria of path planning games, given the best response mathematical programs. Finally, we develop a novel mathematical program for computing a cooperative multi-agent path plan which optimally trades off efficiency and safety among all agents—that is, again, taking the economic perspective on the multi-agent path planning problem.

We numerically investigate path planning games through several case studies involving two and three agents. Our central observation is that as safety becomes more important to agents, a large gap opens up between safety achieved by a socially optimal and Nash equilibrium outcomes; in other words, Nash equilibria exhibit significantly more collisions than desirable by all agents. The main reason for this is that while each agent is concerned with safety, they only account in their objective for the impact of collisions on themselves, and not on other agents who crash along with them.

Our observation about safety consequences of path planning games raises a concern as we look towards the future of autonomous vehicles interacting in populated environments, particularly as they tend to be designed primarily in service of their individual ends, rather than those of the entire autonomous and non-autonomous vehicle ecosystem.

2 Related Work

One common paradigm for studying multi-agent path planning problems is by considering cooperative path planning involving multiple agents. For example, Shen et al. [22] studied cooperative path planning in UAV control system, while LaValle [15] presented an algorithm for applying path planning with stochastic optimal control.

Game theoretic problems related to path planning have been considered from several perspectives. Closest to traditional path planning are zero-sum models of games against nature in which agents are designed to be robust against adversarial uncertainty in the environment [8, 7]. Classic approaches consider rules of interaction and negotiation among self-interested agents, including planning agents [19, 13, 14]. Loosely related also is the extensive literature on multi-agent learning, in which multiple agents repeatedly interact in strategic scenarios in which rewards and dynamics depend on all agents (often modeled as stochastic games) [23].

Another important class of game theoretic models related to path planning are *routing games*. The routing games, as a framework for modeling routing traffic in a large communication network, were first informally discussed by Pigou [18]. This model was first formally defined by Wardrop [24] based on a flow network under the non-atomicity assumption. Therefore, equilibrium flows in non-atomic selfish routing games are often called *Wardrop equilibria*. Since then, a number of fundamental results for the non-atomic routing games have been proved by various researchers, such as the existence and uniqueness of equilibrium flows [3], first-order conditions for convex programming problem [4], and the theory of general non-cooperative non-atomic games [21]. The seminal work by Roughgarden and Tardos [20] first characterized the gap between centralized and decentralized control in multi-agent routing problems, formalized as the price of anarchy, or ratio of socially optimal to worst-case equilibrium outcomes. Their work explained the principles behind a broad class of counter-intuitive phenomena, such as Braess's Paradox [6].

Both routing games and path planning games investigate the competition among agents during their navigation tasks (e.g. passing through bottlenecks). However, in routing games, the state space is a graph-based structure, and the cost of competition is modeled by a set of latency functions without considering the agents' dynamics, while path planning games consider the problem at higher fidelity, with a continuous state space where the latency is caused by the interaction among agents. Moreover, our model of path planning games allows us to explicitly study the tradeoff agents make between performance and safety, an issue not considered in routing games.

3 Model

We describe the problem by first introducing the model of agents' motions, and then formulating the path planning game.

Consider a state space $\mathcal{X} = \mathbb{R}^n$. We represent an agent i by a polyhedron described by a collection of M_i hyperplanes: $P_i = \{a_{ij}^T x \leq b_{ij}, j \in \{0, ..., M_i\}\}$. Each agent polyhedron P_i contains a point $r_i \in \mathcal{X}$ called the *reference* which rigidly attaches to the polyhedron such that the state of an agent can be determined by the position of its *reference*. We assume that agents move in discrete time, and a control input $u_{it} \in \mathcal{U}_{it} \subset \mathbb{R}^m$ applied to the ith agent at time t moves the agent from state $r_{i,t} \in \mathcal{X}$ at time t to state $r_{i,t+1} \in \mathcal{X}$ at time $t+1$ according to a linear stochastic dynamic model

$$r_{i,t+1} = A_i r_{it} + B_i u_{it} + \omega_i, \tag{1}$$

where $A_i \in \mathbb{R}^{n\times n}, B_i \in \mathbb{R}^{n\times m}$, and $\omega_i \sim \mathcal{N}(0, \Sigma_i)$ is the process noise for ith agent at time t following an n-dimension zero-mean Gaussian distribution with a covariance matrix Σ_i.

For each agent we are given its initial placement $r_0 \in \mathcal{X}$ (i.e., where the agent starts) and a goal $r_{goal} \in \mathcal{X}$ which the agent needs to reach. Let $r_{i,0:T} =< r_{i0}, ..., r_{iT} >$ be a state sequence of the (reference point of the) ith agent from time 0 to T and $u_{i,0:T} =< u_{i0}, ..., u_{iT} >$ be a corresponding control sequence. However, once the agent reaches its goal, it remains there deterministically, and has no effect on other agents. We aim to find the optimal control sequence for the ith agent in this stochastic motion model, with the following criteria in mind:

1. After applying the resulting control sequence, the expected terminal position of the ith agent is $r_{i,goal}$,
2. the upper bound of the probability that the ith agent collides with other agents should be minimized, and
3. the agent reaches the goal in as few time steps as possible.

For the moment, we allow no feedback from observed state to control; we relax this restriction below.

Path Planning Game: Given these models of individual agents, we define a *path planning game* by a collection of N agents, with each agent i's action space comprised of all possible control sequences, $\prod_{t=0}^{T} \mathcal{U}_{it}$. In this game, each agent aims to compute an optimal control sequence, given the behavior of others, trading off two objectives: efficiency, or the number of times steps it takes to reach the goal, and safety, or the probability of collision. To formalize, let T_i be the expected number of times steps to reach the goal (if no collision occurs), and G_i the *safety margin*, related to the upper bound on the probability of collision as discussed below. An agent i's objective is then

$$J_i(u_{i,0:T_{max}}, u_{-i,0:T_{max}}) = \lambda T_i + (1 - \lambda) G_i, \tag{2}$$

What makes this a game is that the safety G_i of an agent i depends on the paths taken by *all agents*, rather than i alone. For example, if two agents are moving towards one another, and directly towards their respective goals, the only way for one of them to avoid collision is to circumnavigate the other, taking a longer path towards the goal. Next, we describe how to define and compute T_i and G_i, and compute a *best response* for a given agent i, fixing behavior of all others.

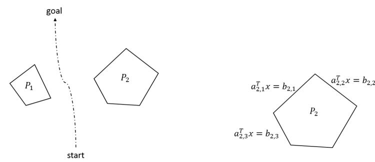
4 Computing an Agent's Best Response

An important subproblem of computing a Nash equilibrium of a path planning game is to compute a best response of an arbitrary agent i when we fix the control policies of all others. We show that calculating agents' best responses in path planning games amounts to a single-agent path planning problem with motion uncertainty. Blackmore et al. [5] previously developed a probabilistic approach for computing a robust optimal path for a robot in the environment with a static obstacle and motion uncertainty via mathematical programming. However, in our context, where an agent trades off efficiency and safety, with stochastic *moving* obstacles (representing other agents), this prior approach is inadequate. In this section we develop a novel method for solving such problems.

4.1 Best Response for a Point-Like Agent

First, consider a simple path planning problem illustrated in Figure 1. In this problem, there is a set of static obstacles and an agent, represented by a point, aiming to find a collision-free minimum-time path from its initial placement to its goal position under motion uncertainty. Assume each obstacle has a given collision volume which can be represented by a polyhedron. To create a mathematical program for solving this problem, two factors need to be taken into account: goal position constraints and collision avoidance constraints.

Formally, let r_t denote the position of an agent at time t with its initial placement r_0 and the goal position r_{goal}. Suppose that the motion dynamics of the agent follows (1) (from which, we remove the index i, since there is only one agent). Assume there are K



(a) (b)

Fig. 1: Single agent path planning with a point-like agent.

obstacles represented by polyhedra $P_n, n = 1, ..., K$, with $P_n = \{x | a_{np}^T x \leq b_{np}, p = 1, \ldots, E_n\}$, where E_n is the number of hyperplanes representing the polyhedron P_n. As before, let T denote the planning horizon (so that the goal must be reached by time T; we assume the horizon is long enough that the goal can be successfully reached even with the obstacles).

Efficiency and Reachability: Let $\{d_0, ..., d_T\}$ denote a collection of binary indicators which indicate whether the agent has reached its goal, i.e., $d_t = 1$ iff $r_t = r_{goal}$. Then, with a large positive number M, the constraints

$$\forall t, ||r_t - r_{goal}|| \leq M(1 - d_t) \tag{3}$$

$$\sum_{t=0}^{T} d_t = 1 \tag{4}$$

make sure that the agent will reach to its goal position sooner or later (and we assume that there exists a feasible solution). Moreover, the number of time steps to reach its goal position can be represented by

$$T = \sum_{t=0}^{T} t \cdot d_t \tag{5}$$

which is one of our objectives (corresponding to T_i, for an agent i above). Since r_t is a random variable, this constrain is stochastic. We approximate it by a deterministic constraint, replacing the position of the agent r_t with its expected position $\overline{r}_t$ in Constraint (3).

Collision Avoidance: Let A denote the event that the agent has a collision, and let $A(n,t), n \in \{1, ..., K\}$ denote the event that the agent collides with the nth obstacle at time step t. We wish to minimize the probability of a collision, $Pr(A)$, or minimize G such that

$$Pr(A) \leq G. \tag{6}$$

The agent has a collision if the agent collides with any of obstacles at any time steps, which is the event

$$A = \bigvee_{t=0}^{T} \bigvee_{n=1}^{K} A(n,t) \tag{7}$$

Then, by the union bound

$$Pr(A) = \Pr\left(\bigvee_{t=0}^{T}\bigvee_{n=1}^{K} A(n,t)\right) \leq \sum_{t=0}^{T}\sum_{n=1}^{K} \Pr(A(n,t)) \leq G \tag{8}$$

$$\Leftarrow [\forall n,t, Pr(A(n,t)) \leq g(n,t)] \wedge [\sum_{t=0}^{T}\sum_{n=1}^{K} g(n,t) = G], \tag{9}$$

where $g(\cdot)$ is *risk allocation* which indicates how the risks are distributed among obstacles and time steps. Next, we consider the event that the agent collides with an obstacle at time step t, which means that the position of the agent is inside the corresponding polyhedron. Thus, collision with the nth obstacle can be described by

$$A(n,t) : \bigwedge_{p=1}^{E_n} a_{np}^T \cdot r_t \leq b_{np} \tag{10}$$

Since the condition (10) including r_t is also stochastic, to convert it into a deterministic one, we consider its probabilistic measure, $\Pr\{A(n,t)\}$. Following (6), our constraints then become

$$\Pr\left\{\bigwedge_{p=1}^{E_n} a_{np}^T \cdot r_t \leq b_{np}\right\} \leq g(n,t). \tag{11}$$

Since a polyhedron is convex, a sufficient condition is,

$$\bigvee_{p=1}^{E_n} \Pr(a_{np}^T \cdot r_t \leq b_{np}) \leq g(n,t). \tag{12}$$

Based on the approach by Blackmore et al. [5], expression (11) can be further simplified using the linear approximation of the upper bound on the probability of collision. First, consider r_t, the position of agent at time step t given its initial placement r_0 and the control sequence $u_{0:t}$, which is a random variable following a Gaussian distribution, $r_t \sim N(\overline{r}_t, \varSigma_t)$, where

$$\overline{r}_t = \sum_{k=0}^{t-1} A^{t-k-1}Bu_k + A^t r_0 \tag{13}$$

and

$$\varSigma_t = \sum_{k=0}^{t-1} A^{t-k-1}\Sigma(A^T)^{t-k-1}. \tag{14}$$

For a single Gaussian random variable $X \sim N(\mu, \sigma^2)$, we can take the inverse Gaussian distribution function at both sides of $\Pr(X < 0) \leq \delta$ and get $u \geq \sqrt{2}\sigma erf^{-1}(1-2\delta)$. Similarly, from $r_t \sim N(\overline{r}_t, \varSigma_t)$, we can get $(a_{np}^T r_t - b_{np}) \sim N(a_{np}^T\overline{r}_t - b_{np}, a_{np}^T\varSigma_t a_{np})$. Then, we take the inverse Gaussian distribution function at both sides of (12), and

$$\bigvee_{p=1}^{E_n} a_{np}^T r_t - b_{np} \geq e(n,t) \tag{15}$$

where $e(n,t) = \sqrt{2a_{np}^T \Sigma_t a_{np}} \cdot erf^{-1}(1 - 2g(n,t))$ and $erf(z) = \frac{2}{\sqrt{\pi}} \int_0^z e^{-t^2} dt$. We call this the *safety margin*, because it expands the margin of obstacles and shrinks the feasible planning domain in order to consider motion uncertainty. Because the motion of the agent after it reaches its goal has no further effect, we add the term $M \sum_{k=0}^{t} d_k$ to these constraints where M is a large positive number.

Define $s(n,t) = erf^{-1}(1 - 2g(n,t))$. Since erf^{-1} is strictly monotonically increasing, we can minimize $\sum_{t=0}^{T} \sum_{n=1}^{K} g(n,t)$ by minimizing

$$G = -\sum_{t=0}^{T} \sum_{n=1}^{K} s(n,t). \tag{16}$$

This is the *safety* portion of an agent's objective (G_i for an agent i above).

A Path Planning Mathematical Program: Our goal is to minimize $J = \lambda T + (1 - \alpha)G$, balancing efficiency and safety using an exogenously specified parameter λ. Combining this objective with the goal and collision avoidance constraints described above, we obtain the following mathematical program for single-agent path planning:

MP1:

$$\min_{u,s,d} \lambda T(d) + (1-\lambda) G(s) \tag{17}$$

s.t.

$$\forall t, u_t \in \mathcal{U}_t \tag{18}$$

$$\forall t, \overline{r}_t = \sum_{k=0}^{t-1} A^{t-k-1} B u_k + A^t r_0 \tag{19}$$

$$\forall t, ||\overline{r}_t - r_{goal}||_1 \leq M \cdot (1 - d_t) \tag{20}$$

$$\forall t, d_t \in \{0, 1\} \tag{21}$$

$$\sum_{t=0}^{T} d_t = 1 \tag{22}$$

$$\forall t \forall n, \bigvee_{p=1}^{E_n} a_{n,p}^T \overline{r}_t > b_{np} + e(n,t) - M \sum_{k=0}^{t} d_k \tag{23}$$

$$\forall t, e(n,t) = s(n,t) \sqrt{a_{np}^T \Sigma_t a_{np}} \tag{24}$$

$$\forall t, \Sigma_t = \sum_{k=0}^{t-1} A^{t-k-1} \Sigma (A^T)^{t-k-1} \tag{25}$$

$$\forall t \forall n, 0 \leq s(n,t) \leq M' \tag{26}$$

One residual concern is that if an agent cannot possibly collide with an nth obstacle at time step t (i.e., if $g(n,t) = 0$), $s(n,t)$ can become unbounded. To address this, we add Constraint (26) which imposes an upper bound M' on $s(\cdot)$, where M' is an appropriate positive number so that $erf(M') \simeq 1$.

Since MP1 is a disjunctive linear program which can be solved by an off-the-shelf linear programming solver. A solution $< u, s(\cdot), d >$ found by MP1 with $d_{T_0} = 1$ means that the agent can reach to its goal position in T_0 time steps with the probability of collision at most $\sum_{t=0}^{T_0} \sum_{n=1}^{K} \frac{1-erf(s(n,t))}{2}$ by applying the control sequence $u_{0:T_0}$.

4.2 Generalization: Feedback Control

Above we considered *open loop* path planning where the control sequence is deterministic and fixed a priori. We now extend our approach to *closed loop (feedback)* control, following the ideas in Geibel and Wysotzki [12] and Oldewurtel et al. [17].

Assume we have a nominal control sequence $\overline{u}_{0:T}$. Then, the feedback control sequence can be obtained by integrating the nominal control sequence and the feedback gain:

$$u_t = \overline{u}_t + K(x_t - \overline{x}_t), \tag{27}$$

where x_t is the observed and $\overline{x}_t$ the predicted position, and K is an exogenous parameter which determines the importance of the error feedback term $(x_t - \overline{x}_t)$. In this approach, $\overline{u}_t$ is computed using the MP1 offline, and the actual control sequence is then generated at runtime by applying (27). As a consequence, the Constraints (25) above become

$$\Sigma_t = \sum_{k=0}^{t-1} (A + BK)^{t-k-1} \Sigma[(A + BK)^T]^{t-k-1}. \tag{28}$$

Notice that when there is no error feedback ($K = 0$) this becomes equivalent to open loop control.

4.3 Collision Avoidance for Polyhedral Agents

Having considered the problem for point-like agents, and then generalizing the approach to consider error feedback, we now generalize the collision avoidance constraints to polyhedral agents.

Consider states of the agent and the nth obstacle, both represented by polyhedra P_t and P_n, respectively. The position of the agents' reference is r_t. Since the reference point rigidly attaches to the agent, let $C = \{x - r_t | x \in P_t\}$ denote the relative region of the agent to its time-dependent reference. When the agent collides with the nth obstacle at time t, we know that $\exists x \in P_t \cap P_n$ (i.e., the intersection of these time-dependent polyhedra is non-empty). Thus, from the point view of the agent, the set of positions of its reference causing collision with the nth obstacle can be represented by $K_n = \{x - c | x \in P_n, c \in C\} = -C \oplus P_n$, where $\oplus$ is the Minkowski addition. Since both C and P_n are polyhedra, K_n is a polyhedron and can be represented by a set of hyperplanes: $K_n = \{x | a_{np}^T x \leq b_{np}, p = 0, ..., E_n\}$, where E_n the number of hyperplanes of K_n. The agent collides with the nth obstacle at time step t if the position of its reference is in K_n, that is, when

$$r_t \in K_n \Leftrightarrow \bigwedge_{p=1}^{E_n} a_{np}^T r_t \leq b_{np}. \tag{29}$$

Comparing (29) with (10), we can see that the problem with polyhedral agents can also be solved via the mathematical program above, if we treat the agent as its reference point, and assign the collision volume K_n to each obstacle.

4.4 Best Response Solver

Our final challenge is to consider the actual best response problem of an arbitrary agent in the path planning game, where all other agents are *moving* (rather than static) obstacles with known stochastic motion policies. We now address this problem, obtaining the final mathematical program for computing a single-agent best response.

Let i denote the agent for whom we are computing a best response, with $-i = \{1, ..., i-1, i+1, ..., N\}$ the set of all others. Let i be represented by a polyhedron P_{it} with reference r_{it} and let $j \in -i$ be represented by P_{jt} with reference r_{jt}. Let C_i denotes the relative region of i to its reference, while C_j denotes the relative region of $j \in -i$ to its reference. Suppose that j reaches its goal position by time step T_j with the corresponding known control sequences $u_{j,0:T_j}$. Then, for each j and t, $K_{ijt} = -C_i \oplus P_{jt}$ is a polyhedron with $K_{ijt} = \{x | a_{ijp}^T x \le b_{ijtp}, p \in \{0, ..., E_{ij}\}\}$ where E_{ij} is the number of hyperplanes related to the shapes of C_i and C_j.

Now we formalize how the control sequence $u_{j,0:T_j}$ of each agent j affects P_{jt} so that we can determine K_{ijt}. From motion dynamics of i and j,

$$r_{it} = \sum_{k=0}^{t-1} A_i^{t-k-1} B_i u_{ik} + A_i^t r_{i0} + \omega_{it} \tag{30}$$

$$\forall j,\ r_{jt} = \sum_{k=0}^{t-1} A_j^{t-k-1} B_j u_{jk} + A_j^t r_{j0} + \omega_{jt} \tag{31}$$

From the perspective of agent i, the motion of agent j can be treated as deterministic if we "migrate" motion uncertainty from j to i so that

$$\begin{aligned}
\forall j, r'_{ijt} &= \sum_{k=0}^{t-1} A_i^{t-k-1} B_i u_{ik} + A_i^t r_{i0} + \omega_{it} - \omega_{jt} \\
\forall j, r'_{jt} &= \sum_{k=0}^{t-1} A_j^{t-k-1} B_j u_{jk} + A_j^t r_{j0}.
\end{aligned} \tag{32}$$

For each j, let $\omega'_{ijt} = (\omega_{it} - \omega_{jt}) \sim N(0, \Sigma_{it} + \Sigma_{jt})$ denote the relative motion uncertainty of i to j at time t. Let

$$\forall j, \Delta r'_{jt} = \sum_{k=0}^{t-1} A_j^{t-k-1} B_j u_{jk} + A_j^t r_{j0} - r_{j0} \tag{33}$$

denote the position shift of agent j at time step t determined by its control sequence $u_{j,0:T_j}$. Then, we obtain the position of K_{ijt} by shifting K_{ij0} by Δr_{jt}. Since $K_{ijt} = \{x | a_{ijp}^T x \le b_{ijtp}\}$, we obtain

$$b_{ijtp} = b_{ij0p} + a_{ijp}^T \cdot \Delta r'_{jt}. \tag{34}$$

Consequently, we obtain the following mathematical program for i's best response:

MP2:

$$\min_{u,s_i(\cdot),d} J_i = \lambda T_i + (1-\lambda) G_i \tag{35}$$

s.t.

$$\forall t, u_{it} \in \mathcal{U}_{it} \tag{36}$$

$$\forall t, \overline{r}_{it} = \sum_{k=0}^{t-1} A_i^{t-k-1} B_i u_{ik} + A_i^t r_{i0} \tag{37}$$

$$\forall t, ||\overline{r}_{it} - r_{i,goal}||_1 \leq M \cdot (1 - d_{it}) \tag{38}$$

$$\forall t, d_{it} \in \{0,1\} \tag{39}$$

$$\sum_{t=0}^{T} d_{it} = 1 \tag{40}$$

$$\forall j \forall t = 0, ..., T_j,$$

$$\bigvee_{p=1}^{E_{ij}} a_{i,j,p}^T \overline{r}_{it} > b_{ij0p} + a_{ijp} \cdot \Delta r'_{jt} + e_{ijt} - M \sum_{k=0}^{t} d_{ik} \tag{41}$$

$$\forall i \forall t, \Sigma_{it} = \sum_{k=0}^{t-1} (A_i + K_i B_i)^{t-k-1} \Sigma_i [(A_i + K_i B_i)^T]^{t-k-1} \tag{42}$$

$$\forall t \forall j, \Delta r'_{jt} = \sum_{k=0}^{t-1} A_j^{t-k-1} B_j u_{jk} + A_j^t r_{j0} - r_{j0} \tag{43}$$

$$\forall j, e_{ijt} = \sqrt{a_{ijp}^T (\Sigma_{it} + \Sigma_{jt}) a_{ijp}} \cdot s_i(j,t) \tag{44}$$

$$\forall t \forall n, 0 \leq s_i(n,t) \leq M' \tag{45}$$

Notice that the constraints (41) are effective only for $t = 0, ..., T_i$, and i is not affected by any j who reached its goal.

5 Finding Equilibria in Path Planning Games

Armed with the best response solvers for each agent i in a path planning game, our goal is to approximate a Nash equilibrium in the resulting game. We do so by applying *best response dynamics* which, if it converges (which it does in our experiments), yields a Nash equilibrium.

Best response dynamics is an asynchronous iterative algorithm in which a single agent i is chosen in each iteration, and we maximize i's utility (i.e., compute its best response) fixing control strategies for all other agents. Best response of an agent i can be calculated as discussed above.

6 Optimal Multi-Agent Path Planning

We now extend the single-agent best response problem to compute an optimal multi-agent path plan. In this case, the control sequences $u_{i,0:T_i}$ of all agents are unknown a priori (as they are being computed jointly). Compared to calculating an agents' best response, we replace the objective of the current agent with the sum of all agents' objectives, i.e., the new objective is $J = \sum_i J_i$, where J_i is the objective of agent i. Moreover, we add constraints analogous to MP2 to make sure that the collision avoidance conditions hold from the perspective of *every agent simultaneously*. We thus obtain the following mathematical program:

MP3:

$$\min_{u,s,d} J = \sum_{i=1}^{N} J_i \tag{46}$$

s.t.

$$\forall i, t, u_{it} \in \mathcal{U}_{it} \tag{47}$$

$$\forall i, t, \overline{r}_{it} = \sum_{k=0}^{t-1} A_i^{t-k-1} B_i u_{ik} + A_i^t r_{i0} \tag{48}$$

$$\forall i, t, ||\overline{r}_{it} - r_{i,goal}||_1 \leq M \cdot (1 - d_{it}) \tag{49}$$

$$\forall i, t, d_{it} \in \{0, 1\} \tag{50}$$

$$\forall i \sum_{t=0}^{T_{max}} d_{it} = 1 \tag{51}$$

$$\begin{aligned} \forall i, t, -i, \bigvee_{p=1}^{E_{i,-i}} a_{i,-i,p}^T \overline{r}_{it} > b_{i,-i,0,p} + a_{i,-i,p} \cdot \Delta r'_{-i,t} \\ + e_{i,-i,t} - M \sum_{k=0}^{t} (d_{ik} + d_{-i,k}) \end{aligned} \tag{52}$$

$$\forall i \forall t, \Sigma_{it} = \sum_{k=0}^{t-1} (A_i + K_i B_i)^{t-k-1} \Sigma_i [(A_i + K_i B_i)^T]^{t-k-1} \tag{53}$$

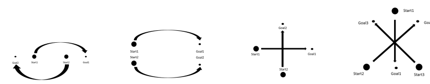
$$\forall i, t, \Delta r'_{it} = \sum_{k=0}^{t-1} A_i^{t-k-1} B_i u_{i,k} + A_i^t r_{i0} - r_{i0} \tag{54}$$

$$\forall i, t \forall -i, e_{i,-i,t} = \sqrt{a_{i,-i,p}^T (\Sigma_{it} + \Sigma_{-i,t}) a_{i,-i,p}} \cdot s_i(t, j)$$

$$\forall i \forall t \forall n, 0 \leq s_i(n, t) \leq M' \tag{55}$$

The term $-M \sum_{k=0}^{t} (d_{ik} + d_{-i,k})$ in Constraints (52) means that an agent will not be affected by other agents who have reached their goal position by time step t, and, conversely, it will not affect the final solution once it reaches its goal position.

7 Experiments



(a) (b) (c) (d)

Fig. 2: Experiment scenarios.

Armed with the techniques for computing both Nash equilibria in path planning games, as well as a socially optimal solution of the corresponding "cooperative" multi-agent planning scenario, we now consider several case studies to understand the impact of self-interested behavior. Specifically, we consider the following 2D scenarios:

- **2 agents with opposing goal positions (Figure 2(a)):** the goal position of each agent is behind the initial placement of the other. In this scenario, the first agent moves from starting coordinate position $(10, 50)$ to goal at position $(95, 50)$, and the second agent moves from $(90, 50)$ to $(5, 10)$.
- **2 agents moving in parallel (Figure 2(b)):** the initial and goal positions of both agents are near one another. In this scenario, the first agent moves from $(10, 70)$ to $(95, 70)$ and the second agent moves from $(10, 35)$ to $(95, 35)$.
- **Intersection with 2 agents (Figure 2(c)):** one agent moves from the bottom to the top of the 2D grid, and the other moves from left to right. In this scenario the first agent moves from $(10, 50)$ to $(90, 50)$ and the second agent moves from $(50, 10)$ to $(50, 90)$.
- **Intersection with 3 agents (Figure 2(d))**: one agent starts at the top of a 2D grid and moves down, while the other two start at southeast and southwest, and move northwest and southeast, respectively. In this scenario the first agent moves from $(50, 90)$ to $(50, 5)$, the second agent moves from $(85, 30)$ to $(11, 73)$, and the third agent moves from $(14, 29)$ to $(90, 73)$.

In each experiment, each agent is represented by a square with each side of length 15 and parallel to either the x or the y axis. The control inputs are 2D velocity vectors and the maximum velocity of agents in both x and y direction is 10 (thus, $A = B = I$ in agents' motion dynamic). Agents' motion is distorted by a Gaussian distribution with the covariance matrix $1.9I$. For each scenario we consider solutions with and without feedback control, where the feedback gain for the latter was chosen to be $K = 0.5$. Throughout, we assume that all players are equally concerned about safety vs. efficiency; formally, all players share the same parameter λ.

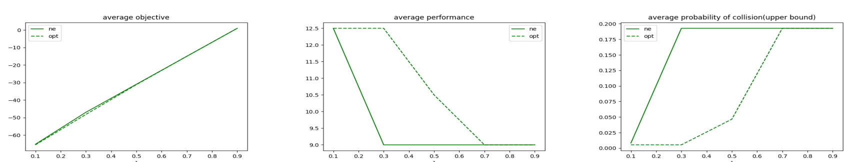
The results are shown in Figures 3-10. In each figure, the horizontal axis is the λ value which represents the importance of safety for both agents, where lower values of λ imply that safety is *more* important. The left plots show the objective value, where lower is better. The middle plots give the time to goal, where lower is, again, better. The right plots show safety margin, where again lower is better. We present average quantities over all agents; the qualitative observations are similar if we consider these at individual agent level.

The first observation is that the difference between socially optimal and equilibrium objective values appears small ((a) plots in Figures 3-10). It is therefore tempting to conclude that equilibrium behavior is similar to socially optimal, but it turns out that this is not the case: in particular, it turns out that the trade-off between efficiency and safety made by the agents in equilibrium is very different from optimal.

Considering next the (b) and (c) columns of the figures, we can observe that systematically performance improves, while safety is often significantly compromised, in equilibrium as compared to a social optimum. The difference is particularly dramatic in the first two scenarios, when the agents are in direct conflict in their quest to reach their respective goals. The gap between optimal and equilibrium safety in the other scenarios tends to be larger for relatively high values of λ.

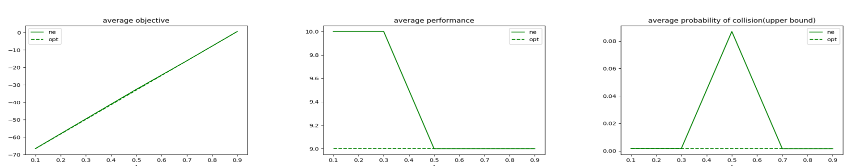
Another general observation we can make is that often the solutions with a feedback controller are closer to optimal, particularly from the perspective of safety. The exceptions involve the intersection scenarios, where the gap is larger for higher values of λ in the feedback controller solution than with the open-loop controller. However, even in these scenarios, the feedback controller yields solutions closer to socially optimal for most values of λ. This is not surprising: since all agents are concerned about safety, they are more able to dynamically adjust to avoid collisions when some feedback about state is available.

To understand why safety is systematically compromised, consider a single agent's incentive. Even though an agent is interested in reaching the goal safely, it does not account for the fact that being involved in a crash *also crashes the other agent*. Thus, in equilibrium safety is compromised relative to social optimum, as agents fail to capture the externalities associated with crashes.



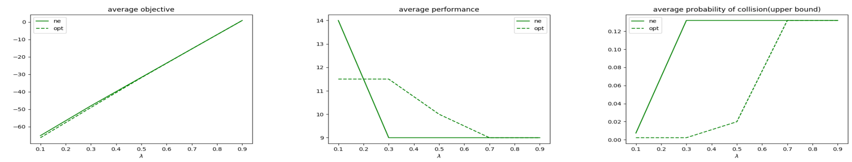
(a) (b) (c)

Fig. 3: Opposing goal positions without the feedback gain($K = 0$).



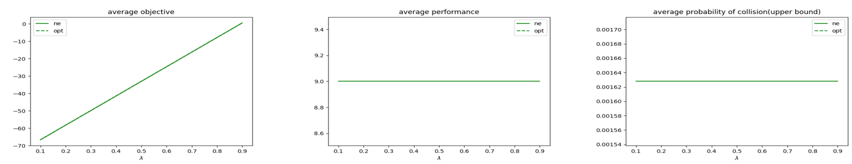
(a) (b) (c)

Fig. 4: Opposing goal positions with the feedback gain($K = 0.5$).



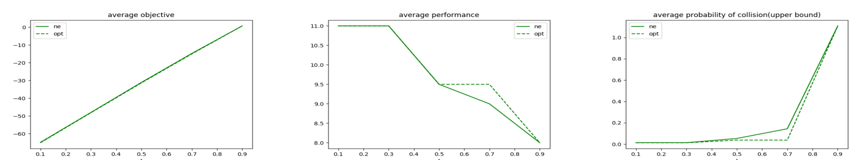
(a) (b) (c)

Fig. 5: Moving in parallel without the feedback gain($K = 0$).



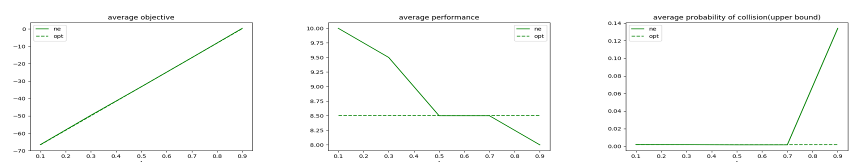
(d) (e) (f)

Fig. 6: Moving in parallel with the feedback gain($K = 0.5$).



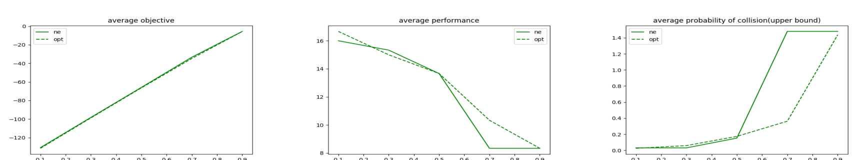
(a) (b) (c)

Fig. 7: Intersection without the feedback gain($K = 0$, 2 agents).



(a) (b) (c)

Fig. 8: Intersection with the feedback gain($K = 0.5$, 2 agents).



(a) (b) (c)

Fig. 9: Intersection without the feedback gain($K = 0$, 3 players).

(a) (b) (c)

Fig. 10: Intersection with the feedback gain($K = 0.5$, 3 players).

Bibliography

[1] Arkin, R.C.: Motor Schema-Based Mobile Robot Navigation (1989)
[2] Auat Cheein, F.A., Carelli, R.: Agricultural robotics: Unmanned robotic service units in agricultural tasks. IEEE Industrial Electronics Magazine 7(3), 48–58 (2013)
[3] Beckmann, M., McGuire, C., Winsten, C.: Studies in the Economics of Transportation (1956)
[4] Bertsekas, D.: Nonlinear programming (1999)
[5] Blackmore, L., Li, H., Williams, B.: A probabilistic approach to optimal robust path planning with obstacles. In: American Control Conference (2006)
[6] Braess, D.: Über ein Paradoxon aus der Verkehrsplanung. Unternehmensforschung Operations Research - Recherche Opérationnelle 12(1), 258–268 (dec 1968)
[7] Chen, M., Zhou, Z., Tomlin, C.J.: Multiplayer reach-avoid games via low dimensional solutions and maximum matching. In: American Control Conferenc. pp. 1444–1449 (2014)
[8] Chen, M., Zhou, Z., Tomlin, C.J.: A path defense approach to the multiplayer reach-avoid game. In: Annual Conference on Decision and Control. pp. 2420–2426 (2014)
[9] Craighead, J., Murphy, R., Burke, J., Goldiez, B.: A survey of commercial & open source unmanned vehicle simulators. In: Proceedings - IEEE International Conference on Robotics and Automation. pp. 852–857 (2007)
[10] DeSouza, G.N., Kak, A.C.: Vision for mobile robot navigation: A survey. IEEE Transactions on Pattern Analysis and Machine Intelligence 24(2), 237–267 (2002)
[11] Elfes, A.: Using occupancy grids for mobile robot perception and navigation. Computer 22(6), 46–57 (1989)
[12] Geibel, P., Wysotzki, F.: Risk-sensitive reinforcement learning applied to control under constraints. J. Artif. Intell. Res. 24, 81–108 (2005)
[13] Jonsson, A., Rovatsos, M.: Scaling up multiagent planning: A best-response approach. In: ICAPS (2011)
[14] Jordán, J., Torreno, A., de Weerdt, M., Onaindia, E.: A better-response strategy for self-interested planning agents. Applied Intelligence pp. 1–21 (2017)
[15] LaValle, S.M.: Robot motion planning: A game-theoretic foundation. Algorithmica 26(3-4), 430–465 (2000)
[16] Mahony, R., Kumar, V.: Aerial robotics and the quadrotor. IEEE Robotics and Automation Magazine 19(3), 19 (2012)
[17] Oldewurtel, F., Jones, C.N., Morari, M.: A tractable approximation of chance constrained stochastic mpc based on affine disturbance feedback. In: IEEE Conference on Decision and Control. pp. 4731–4736 (2008)
[18] Pigou, A.: The economics of welfare, 1920. McMillan&Co., London (1932)
[19] Rosenschein, J.S., Zlotkin, G.: Rules of encounter: designing conventions for automated negotiation among computers. MIT press (1994)
[20] Roughgarden, T., Tardos, E.: How bad is selfish routing? Proceedings 41st Annual Symposium on Foundations of Computer Science 49(2), 1–26 (2000)
[21] Schmeidler, D.: Equilibrium points of nonatomic games. Journal of Statistical Physics 7(4), 295–300 (1973)
[22] Shen, D., Chen, G., Cruz, J.B., Blasch, E.: A game theoretic data fusion aided path planning approach for cooperative UAV ISR. In: Aerospace Conference. pp. 1–9 (2008)
[23] Stone, P., Veloso, M.: Multiagent systems: A survey from a machine learning perspective. Autonomous Robots 8(3), 345–383 (2000)
[24] Wardrop, J.G.: Road paper. some theoretical aspects of road traffic research. Proceedings of the institution of civil engineers 1(3), 325–362 (1952)