

From Low to High Order Motion Planners : Safe Robot Navigation Using Motion Prediction and Reference Governor

Aykut İşleyen¹, Graduate Student Member, IEEE, Nathan van de Wouw², Fellow, IEEE,
and Ömür Arslan³, Member, IEEE

Abstract—Safe navigation around obstacles is a fundamental challenge for highly dynamic robots. The state-of-the-art approach for adapting simple reference path planners to complex robot dynamics using trajectory optimization and tracking control is brittle and requires significant replanning cycles. In this letter, we introduce a novel feedback motion planning framework that extends the applicability of low-order (e.g. position-/velocity-controlled) reference motion planners to high-order (e.g., acceleration-/jerk-controlled) robot models using motion prediction and reference governors. We use predicted robot motion range for safety assessment and establish a continuous bidirectional interface between high-level planning and low-level control via a reference governor. We describe the generic fundamental building blocks of our feedback motion planning framework and give specific example designs for motion control, prediction, and reference planning. We prove the correctness of our planning framework and demonstrate its performance in numerical simulations. We conclude that accurate motion prediction is crucial for closing the gap between high-level planning and low-level control.

Index Terms—Collision avoidance, feedback motion planning, integrated planning and control, motion and path planning.

I. INTRODUCTION

SAFE and smooth robot motion is fundamental for many autonomous systems, and people interacting with them. Kinodynamic motion planning of dynamically feasible and safe trajectories is known to be computationally hard for many robotic systems [1], [2], because determining safety of highly dynamic robot systems is a challenge [3]. State-of-the-art smooth motion planning approaches [4] often start with a simple reference path planner [5], and then build a dynamically feasible and safe robot trajectory (i.e., a time-parametrized path) around a reference path using trajectory optimization [6]. However, such trajectory planning methods often suffer from significant replanning cycles in practice due to precise trajectory tracking requirements [7], [8].

Manuscript received 24 February 2022; accepted 22 June 2022. Date of publication 18 July 2022; date of current version 27 July 2022. This letter was recommended for publication by Associate Editor J. D. Hernandez and Editor H. Kurniawati upon evaluation of the reviewers' comments. (Corresponding author: Aykut İşleyen.)

The authors are with the Department of Mechanical Engineering, Eindhoven University of Technology, 5600 MB Eindhoven, The Netherlands (e-mail: a.isleyen@tue.nl; n.v.d.wouw@tue.nl; o.arslan@tue.nl).

This letter has supplementary downloadable material available at <https://doi.org/10.1109/LRA.2022.3191786>, provided by the authors.

Digital Object Identifier 10.1109/LRA.2022.3191786

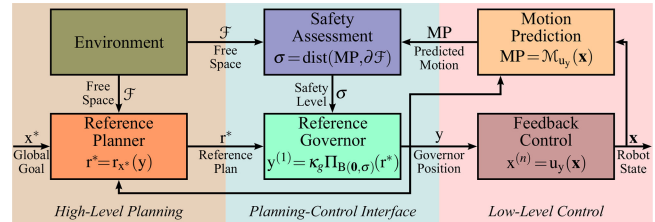


Fig. 1. Safe feedback motion planning using motion prediction and reference governor enables a bidirectional safety interface between high-level planning and low-level control.

In this letter, we propose a novel provably correct feedback motion planning framework that extends safety and navigation properties of low-order (e.g. position-/velocity-controlled) reference motion planners to high-order (e.g., acceleration-/jerk-controlled) robot dynamics using a continuous bidirectional interface between high-level planning and low-level control based on a reference governor and safety assessment of predicted robot motion, as illustrated in Fig. 1.

A. Motivation and Relevant Literature

Safe and smooth robot motion is essential for bringing robots from factories to our everyday lives, because autonomous robot operation around people requires safety guarantees, and jerky robot motion does not only cause system failures and malfunctions, but also causes discomfort for people. Most existing safe and smooth motion planning methods follow a two-step approach: first find a simple piecewise linear path using a standard off-the-shelf path planner [5], and then perform path smoothing and time parametrization using trajectory optimization in order to satisfy safety and system requirements [6], [9], [10]. Unfortunately, the open-loop nature of such trajectory planning methods often requires many replanning cycles to ensure safety by closing the planning loop [7], [8]. Robustified trajectory planning and tracking based on the worst-case tracking error bound suffers less from replanning, but comes with conservatism [11]. In this letter, we introduce a new feedback motion planning framework for safe and smooth robot motion design by establishing a continuous bidirectional safety interface between high-level reference planning and low-level motion control via motion prediction and reference governors. Our approach avoids replanning via reference governors and induces less conservatism by using accurate motion prediction.

Reference governors are originally designed for constrained control of dynamical systems to separately handle the issues of stability and constraint satisfaction [12]–[14]. In robotics, reference governors are applied for safe robot navigation to separately address global navigation, safety, and stability, and safety requirements at different stages by high-level planning and low-level control [15]. Reference governors are successfully demonstrated for safe navigation of second-order robotic systems using total energy [15] and directed Lyapunov ellipsoids [16], [17]. These existing approaches indirectly suggest that motion prediction and safety assessment is fundamental in governed robot motion design in their very specific constructions. In this letter, we present a unified feedback motion planning framework based on motion prediction and reference governors, describe generic fundamental building blocks for governed robot motion design, provide new example designs, and systematically investigate the role of motion prediction and system order on robot motion. At a more conceptual level, our newly introduced notion of motion range prediction allows one to effectively determine (inevitable) collision states of high-order robotic systems which is a known bottleneck in kinodynamic motion planning [3].

B. Contributions and Organization of the Letter

This letter introduces a new unifying feedback motion planning framework that closes the gap between traditionally separately studied high-level planning and low-level control by establishing a continuous bidirectional planning-control interface based on motion prediction, safety assessment, and reference governors. In Section III, we present fundamental motion design elements of our motion planning framework, describe its working principles, and analyze its stability and safety properties. In Section IV, we provide example designs for low-level motion control and prediction, and high-level reference planning, and demonstrate the use of Lyapunov ellipsoids and Vandermonde simplexes for robot motion prediction under proportional higher-order derivative (PhD) control. In Section V, we systematically investigate the role of motion prediction on robot motion in numerical simulations. We conclude in Section VI with a summary of our contributions and future work.

II. FROM LOW TO HIGH ORDER MOTION PLANNERS: PROBLEM FORMULATION

For ease of exposition, we consider a disk-shaped robot of radius $\rho \in \mathbb{R}_{>0}$, centered at position $\mathbf{x} \in \mathcal{W}$, operating in a known static closed compact environment $\mathcal{W} \subseteq \mathbb{R}^d$ that is cluttered with a collection of obstacles represented by an open set $\mathcal{O} \subset \mathbb{R}^d$, where $d \geq 2$. Hence, the robot's free space, denoted by \mathcal{F} , of collision-free positions is given by

$$\mathcal{F} := \{\mathbf{x} \in \mathcal{W} \mid \mathcal{B}(\mathbf{x}, \rho) \subseteq \mathcal{W} \setminus \mathcal{O}\}, \quad (1)$$

where $\mathcal{B}(\mathbf{x}, \rho) := \{\mathbf{y} \in \mathbb{R}^d \mid \|\mathbf{y} - \mathbf{x}\| \leq \rho\}$ is the Euclidean closed ball centered at \mathbf{x} with radius ρ , and $\|\cdot\|$ denotes the standard Euclidean norm for both vectors and matrices. To ensure global navigation between any start and goal positions in \mathcal{F} , we assume that the free space \mathcal{F} is path-connected.

We also assume that the robot motion is fully actuated and described by an n th-order differential equation of the form¹

$$\mathbf{x}^{(n)} = \mathbf{u}(\mathbf{x}) = \mathbf{u}\left(\mathbf{x}^{(0)}, \mathbf{x}^{(1)}, \dots, \mathbf{x}^{(n-1)}\right), \quad (2)$$

where $\mathbf{x}^{(k)} = \frac{d^k}{dt^k} \mathbf{x}$ denotes the k th time derivative of the robot position \mathbf{x} , and $\mathbf{u} : (\mathbb{R}^d)^n \rightarrow \mathbb{R}^d$ is a feedback control policy, and $\mathbf{x} = (\mathbf{x}^{(0)}, \mathbf{x}^{(1)}, \dots, \mathbf{x}^{(n-1)}) \in (\mathbb{R}^d)^n$ denotes the robot's dynamical state consisting of its position, velocity, and so on. Accordingly, one can determine if the robot is in motion or not as follows.

Definition 1: (Zero Motion) An n th-order robot is said to be in the *zero-motion* state iff its velocity and higher-order variables are zero, i.e., $\mathbf{x}^{(k)} = \mathbf{0}$ for all $k = 1, \dots, n-1$, where $\mathbf{0}$ is a vector of all zeros of the appropriate size. Hence, any robot state $\mathbf{x} \in \mathbb{R}^d \times \{\mathbf{0}\}^{n-1}$ is a zero-motion state.

For example, for the second-order dynamics, zero-motion corresponds to zero-velocity. Note that zero-motion is different than being stationary which means, in addition to the condition for zero motion, $\mathbf{u}(\mathbf{x}) = \mathbf{0}$.

For the simple fully-actuated kinematic (position- and velocity-controlled) robot model, the global navigation problem can be solved effectively using off-the-shelf motion planning algorithms [5]. However, kinodynamic motion planning of complex high-order robotic systems is an open research challenge because collision and safety verification of highly dynamical robotic systems is hard [2]. As a new approach for closing the gap between kinematic and kinodynamic motion planning, we consider *provably-correct extensions of feedback motion planners* from simple (i.e., first-order) to complex (i.e., n th-order) robot dynamics that preserve and augment stability and safety properties:

From Low to High Order Safe Motion Planners: Given a Lipschitz continuous vector field planner $\mathbf{r}_{\mathbf{x}^*} : \mathcal{D} \rightarrow \mathbb{R}^d$ for the first-order robot dynamics

$$\dot{\mathbf{x}}^{(1)} = \mathbf{r}_{\mathbf{x}^*}(\mathbf{x}), \quad (3)$$

that asymptotically steers all robot positions in its positively invariant collision-free domain $\mathcal{D} \subseteq \mathcal{F}$ towards a safe goal location $\mathbf{x}^* \in \mathcal{D}$, a correct extension of the reference motion planner $\mathbf{r}_{\mathbf{x}^*}$ for the n th-order robot dynamics is the design of a feedback motion planner $\mathbf{u}_{\mathbf{r}_{\mathbf{x}^*}} : (\mathbb{R}^d)^n \rightarrow \mathbb{R}^d$ of the form

$$\mathbf{x}^{(n)} = \mathbf{u}_{\mathbf{r}_{\mathbf{x}^*}}(\mathbf{x}) = \mathbf{u}_{\mathbf{r}_{\mathbf{x}^*}}\left(\mathbf{x}^{(0)}, \mathbf{x}^{(1)}, \dots, \mathbf{x}^{(n-1)}\right), \quad (4)$$

that asymptotically brings all collision-free zero-motion initial robot states² in $\mathcal{D} \times \{\mathbf{0}\}^{n-1}$ to the goal location \mathbf{x}^* while ensuring no collision along the way, i.e., the robot trajectory $\mathbf{x}(t)$ stays in the free space \mathcal{F} for all future times $t \geq 0$.

Below, we introduce a general framework with examples for provably correct extensions of feedback motion planners from low-order to high-order robot dynamics using reference governors and risk assessment of predicted robot motion.

¹For example, one can consider a nonholonomic robotic system controlled via feedback linearization [18] or differential flatness [19].

²This requirement on the initial robot states can be extended to include other collision-free robot states as defined in Corollary 1.

III. FROM LOW TO HIGH ORDER MOTION PLANNERS: GENERAL FRAMEWORK

In this section, we describe the generic building blocks and the working principle of our motion planning framework for extending the applicability of a first-order reference planner to the n th-order robot dynamics that are stabilized by a standard feedback motion control policy. We realize a bidirectional interface between the high-level reference motion planner and the low-level feedback motion control via a reference governor system using motion range prediction and safety assessment, as illustrated in Fig. 1. We also analyze important stability and safety properties of our general extension framework to demonstrate its correctness.

A. Fundamental Motion Planning Elements

Our motion planning framework consists of five building elements: reference motion planner, feedback motion control, motion prediction, safety assessment, and reference governor, whose specific roles and requirements are presented below.

1) *Reference Motion Planner*: A reference motion planner for the first-order and fully-actuated robot dynamics in (3) is a Lipschitz continuous vector field planner $r_{x^*} : \mathcal{D} \rightarrow \mathbb{R}^d$ associated with a goal location $x^* \in \mathcal{D}$ in its collision-free domain $\mathcal{D} \subseteq \mathcal{F}$ such that

- (Positive Invariance & Boundary Avoidance)³ r_{x^*} is inward pointing on the boundary $\partial\mathcal{D}$ of its domain \mathcal{D} .
- (Global Asymptotic Stability) r_{x^*} has a unique stable point at x^* whose domain of attraction includes \mathcal{D} .

In other words, for the velocity-controlled robot model, the reference motion planner r_{x^*} asymptotically brings all robot positions in its positively-invariant domain \mathcal{D} towards the goal x^* while avoiding its boundary and so collisions [15].

2) *Feedback Motion Control*: A feedback motion control policy $u_y : (\mathbb{R}^d)^n \rightarrow \mathbb{R}^d$ for the n th-order robot dynamics

$$\dot{x}^{(n)} = u_y(\mathbf{x}) = u_y(x^{(0)}, x^{(1)}, \dots, x^{(n-1)}), \quad (5)$$

is a Lipschitz continuous controller that is parametrized with a desired robot position $y \in \mathbb{R}^d$ at which the closed-loop system is globally asymptotically stable with zero-motion (Definition 1), i.e., $u_y(y, \mathbf{0}, \dots, \mathbf{0}) = \mathbf{0}$. As a controller choice, one can consider any standard feedback motion controller as long as its motion range can be accurately bounded in terms of the robot state \mathbf{x} and the goal y , as discussed next.

3) *Motion Range Prediction*: For the n th-order robot model that moves towards a given goal $y \in \mathbb{R}^d$ under the feedback motion control u_y , a *motion range prediction*, denoted by $\mathcal{M}_{u_y}(\mathbf{x})$, is a closed set that bounds the robot motion trajectory $\mathbf{x}(t)$ starting at $t = 0$ from an initial state $\mathbf{x} \in (\mathbb{R}^d)^n$ for all future times $t \geq 0$, i.e.,

$$\mathbf{x}(t) \in \mathcal{M}_{u_y}(\mathbf{x}) \quad \forall t \geq 0, \quad (6)$$

and is bounded by a Euclidean ball centered at y as

$$\mathcal{M}_{u_y}(\mathbf{x}) \subseteq B(y, \eta \|\mathbf{x} - (y, \mathbf{0}, \dots, \mathbf{0})\|) \quad \forall \mathbf{x} \in (\mathbb{R}^d)^n, \quad (7)$$

where $\eta \in \mathbb{R}_{>0}$ is a fixed positive constant. Note that the bound on motion range prediction in (7) implies that the motion prediction asymptotically converges to a single point at y since the

³The boundary avoidance property is required for ensuring that the robot-governor system can asymptotically follow a reference motion planner without getting stuck along the way (see the proof of Proposition 2).

feedback motion control u_y is globally asymptotically stabilizing the zero-motion position y , i.e.,

$$\lim_{t \rightarrow \infty} \mathcal{M}_{u_y}(\mathbf{x}) = \{y\}. \quad (8)$$

Hence, a bounded motion range prediction in (7) ensures that the (collision) distance of the motion range prediction $\mathcal{M}_{u_y}(\mathbf{x})$ to the free space boundary $\partial\mathcal{F}$ might be zero only for a finite amount of time for any goal $y \in \mathcal{F}$ in the free space interior \mathcal{F} , which is crucial to avoid undesired critical points, as later discussed in the proof of Proposition 2.

In general, determining collision-free states of high-order dynamical systems is a difficult task in kinodynamic motion planning [2]. The availability of a motion range prediction allows one to effectively identify certain collision-free states of n th-order dynamical systems because having motion range prediction in the free space implies safe robot motion, i.e.,

$$\mathcal{M}_{u_y}(\mathbf{x}) \subseteq \mathcal{F} \implies \mathbf{x}(t) \in \mathcal{F} \quad \forall t \geq 0. \quad (9)$$

Corollary 1: A robot state $\mathbf{x} \in (\mathbb{R}^d)^n$ is collision free under the control policy u_y if there exists a goal position $y \in \mathcal{F}$ relative to which the associated motion range prediction is in the free space, i.e., $\mathcal{M}_{u_y}(\mathbf{x}) \subseteq \mathcal{F}$.

Accordingly, we shall continuously monitor the safety of the closed-loop robot motion with respect to its goal position by measuring the distance of the predicted motion range to the free space boundary as described below.

4) *Safety Assessment*: Given a motion range prediction $\mathcal{M}_{u_y}(\mathbf{x})$ for the closed-loop robot dynamics in (5), the safety level of the robot's trajectory $\mathbf{x}(t)$ starting from an initial state $\mathbf{x} = (x^{(0)}, x^{(1)}, \dots, x^{(n-1)}) \in (\mathbb{R}^d)^n$ towards a given goal $y \in \mathbb{R}^d$ is defined as the minimum distance between the predicted motion range and the free space boundary as

$$\sigma(\mathbf{x}, y) := \text{dist}(\mathcal{M}_{u_y}(\mathbf{x}), \partial\mathcal{F}), \quad (10)$$

$$:= \begin{cases} \min_{\substack{a \in \mathcal{M}_{u_y}(\mathbf{x}) \\ b \in \partial\mathcal{F}}} \|a - b\|, & \text{if } \mathbf{x}^{(0)} \in \mathcal{F}, \\ 0, & \text{otherwise,} \end{cases} \quad (11)$$

where $\partial\mathcal{F}$ denotes the boundary of the free space \mathcal{F} . Here, a safety level of zero means unsafe motion; and the higher the safety level is the safer the motion. Note that we consider being exactly on the boundary of the free space to be unsafe although it is, by definition in (1), free of collisions.

A critical requirement of the safety level measure for our stability analysis performed below is that $\sigma(\mathbf{x}, y)$ is a locally Lipschitz continuous function of the robot state \mathbf{x} and the goal y . For example, $\sigma(\mathbf{x}, y)$ is locally Lipschitz continuous if the motion range prediction can be expressed as an affine transformation of some fixed sets (e.g., the unit ball/simplex) based on a smooth function of the robot state \mathbf{x} and goal y .

Lemma 1: (Lipschitz Continuity of Minimum Set Distance) The minimum distance between two compact sets under an affine transformation is Lipschitz continuous w.r.t. the affine transformation parameters, i.e., for any $X \subset \mathbb{R}^m$ and $Y \subset \mathbb{R}^n$

$$|\text{dist}(f_{\mathbf{A}, \mathbf{b}}(X), Y) - \text{dist}(f_{\mathbf{A}, \mathbf{b}'}(X), Y)| \leq \|\mathbf{b} - \mathbf{b}'\|,$$

$$|\text{dist}(f_{\mathbf{A}, \mathbf{b}}(X), Y) - \text{dist}(f_{\mathbf{A}', \mathbf{b}}(X), Y)| \leq \|\mathbf{A} - \mathbf{A}'\| \max_{\mathbf{x} \in X} \|\mathbf{x}\|,$$

where $\text{dist}(X, Y) = \min_{x \in X, y \in Y} \|x - y\|$ is the minimum set distance, and $f_{A,b}(x) := Ax + b$ is an affine transformation parametrized by $A \in \mathbb{R}^{n \times m}$ and $b \in \mathbb{R}^n$.

Proof: See the supplementary material in [20]. ■

5) *Reference Governor:* A reference governor is a first-order dynamical system with a position state $y \in \mathbb{R}^d$ that follows a reference motion planner $r_{x^*} : \mathcal{D} \rightarrow \mathbb{R}^d$ towards a goal position $x^* \in \mathcal{D} \subseteq \mathcal{F}$ as close as possible, based on the safety level $\sigma(x, y)$ of the predicted robot motion starting from state $x \in (\mathbb{R}^d)^n$ towards the governor position y . We design the reference governor dynamics as follows:

$$y^{(1)} = \kappa_g \Pi_{B(0, \sigma(x, y))}(r_{x^*}(y)), \quad (12a)$$

$$= -\kappa_g (y - \Pi_{B(y, \sigma(x, y))}(y + r_{x^*}(y))), \quad (12b)$$

$$= \kappa_g \min(\sigma(x, y), \|r_{x^*}(y)\|) \frac{r_{x^*}(y)}{\|r_{x^*}(y)\|}, \quad (12c)$$

where $\kappa_g > 0$ is a fixed control gain for the governor, $\Pi_A(b) := \arg \min_{a \in A} \|a - b\|$ denotes the metric projection of a point b onto a closed set A , and $B(0, \sigma)$ is the Euclidean ball centered at the origin with radius $\sigma \geq 0$. This design ensures that the governor is only allowed to move according to the reference planner r_{x^*} if the robot's motion relative to the governor is predicted to be safe, i.e., $\sigma(x, y) = \text{dist}(\mathcal{M}_{u_y}(x), \partial\mathcal{F}) > 0$. Also note that the right-hand side of the reference governor dynamics in (12c) is Lipschitz continuous since both the safety level $\sigma(x, y)$ and the reference planner $r_{x^*}(y)$ are assumed to be Lipschitz.

B. Working Principle of High Order Motion Planners

The proposed feedback motion planning framework consists of a low-level inner control loop and a high-level outer planning loop that continuously and bidirectionally interact with each other via their shared element — the reference governor, see Fig. 1, to transfer the global navigation properties of the first-order reference motion planner to the n th-order robot dynamics while ensuring safety and stability. The working principles of these control and planning loops and their interaction can be conceptually summarized as follows:

- *Low-Level Inner Control Loop:* At the low level, the feedback motion control $u_y(x)$ constantly tries to stabilize the robot state x at the (potentially changing) governor position y with zero-motion, while the safety level $\sigma(x, y)$ of the resulting robot motion is continuously monitored using the predicted robot motion range $\mathcal{M}_{u_y}(x)$ with respect to the governor position y .
- *High-Level Outer Planning Loop:* At the high level, the reference governor tries to follow the reference motion planner r_{x^*} as close as possible, based on the safety level $\sigma(x, y)$ of the current robot state x relative to the governor position y , so that the governor asymptotically reaches to the global goal x^* of the reference plan while ensuring safe robot motion.
- *Low- & High-Level Control and Planning Interaction:* Hence, the reference governor defines a continuous bidirectional interface between the inner control loop and the outer planning loop such that the high-level reference motion plan is adjusted according to the safety assessment of predicted low-level robot motion, and then transferred from the outer planning loop to the inner control loop to ensure safe and stable robot motion.

Finally, it is convenient to write the overall dynamics of the robot-governor system using the closed-loop robot dynamics in (5) and the governor dynamics in (12) as

$$\dot{x}^{(n)} = u_y(x), \quad (13a)$$

$$\dot{y}^{(1)} = \kappa_g \Pi_{B(0, \sigma(x, y))}(r_{x^*}(y)), \quad (13b)$$

where r_{x^*} is a first-order reference dynamics and $\sigma(x, y)$ is the robot's safety level relative to the governor as defined in (10). This clearly shows the strong coupling between the robot and the governor.

C. Safety and Stability Properties

In this part, to prove the correctness of our motion planning framework, we first show that the robot-governor dynamics in (13) result in a safe motion for both the n th-order robot and the first-order governor, and then show that they both asymptotically converge to zero-motion at the global goal x^* by following the first-order reference motion planner.

Proposition 1: (Safety) Starting at $t = 0$ from any collision-free robot state $x(0) \in (\mathbb{R}^d)^n$ relative to a collision-free governor position $y(0) \in \mathcal{D} \subseteq \mathcal{F}$ in the sense of Corollary 1, under the robot-governor dynamics in (13), the robot's trajectory $x(t)$ and the governor trajectory $y(t)$ stays collision free in \mathcal{F} and \mathcal{D} , respectively, for all future times $t \geq 0$, i.e.,

$$\mathcal{M}_{u_y}(x(0)) \subseteq \mathcal{F}, \quad y(0) \in \mathcal{D} \subseteq \mathcal{F} \implies \begin{matrix} x(t) \in \mathcal{F}, \\ y(t) \in \mathcal{D} \subseteq \mathcal{F} \end{matrix} \quad \forall t \geq 0. \quad (14)$$

Proof: The safety of the governor follows from the positive invariance of the collision-free domain \mathcal{D} of the reference planner r_{x^*} because the governor dynamics in (12c) equal to a non-negatively scaled version of the reference plan r_{x^*} which ensures that \mathcal{D} is also positively invariant under (13b).

The safety of the robot relative to the governor (see Corollary 1) can be shown in two steps. If the governor moves, i.e., $y^{(1)}(t) \neq 0$ in (13b), then the positive safety level $\sigma(x(t), y(t)) > 0$ implies $\mathcal{F} \supset \mathcal{M}_{u_{y(t)}}(x(t)) \ni x(t)$. Otherwise (i.e., if the governor is stationary), the safety of the robot trajectory is verified either at the start at $t = 0$ or just before the governor becomes stationary where we have $\sigma(x(t), y(t)) = 0$, but $\mathcal{M}_{u_{y(t)}}(x(t)) \subseteq \mathcal{F}$. Hence, we have by definition (6) that $x(t) \in \mathcal{M}_{u_y}(x(0))$ for all $t \geq 0$. ■

Although the motion range prediction is not necessarily in the free space for all times, it is useful to observe the motion prediction is inside the interior $\mathring{\mathcal{F}}$ of the free space when the governor moves since the safety level is strictly positive, i.e.,

$$y^{(1)}(t) \neq 0 \iff \sigma(x(t), y(t)) > 0 \iff \mathcal{M}_{u_{y(t)}}(x(t)) \subset \mathring{\mathcal{F}}. \quad (15)$$

Proposition 2: (Stability) Starting from any collision-free robot state $x(0) \in (\mathbb{R}^d)^n$ and collision-free governor position $y(0) \in \mathcal{D}$ with positive safety level $\sigma(x(0), y(0)) > 0$, under the robot-governor dynamics in (13), both the robot and the governor asymptotically converge to zero motion at the global goal x^* of the reference motion plan r_{x^*} , i.e.,

$$\lim_{t \rightarrow \infty} \dot{x}^{(0)}(t) = \lim_{t \rightarrow \infty} \dot{y}^{(0)}(t) = x^*, \quad (16)$$

$$\lim_{t \rightarrow \infty} \dot{x}^{(i)}(t) = \lim_{t \rightarrow \infty} \dot{y}^{(1)}(t) = 0, \quad \forall i = 1, \dots, (n-1). \quad (17)$$

Proof: The existence and uniqueness of the robot's and the governor's trajectory follows from the local Lipschitz continuity requirement of the feedback motion control u_y , the reference motion plan r_{x^*} , and the safety level $\sigma(x, y)$ over the compact free space \mathcal{F} [21].

It follows from (15) that the initial positive safety level $\sigma(x, y) > 0$ ensures that the governor is inside the interior $\mathring{\mathcal{F}}$ of the free space \mathcal{F} . Hence, since the reference motion planner (and its nonnegative scaling in (13b)) is inward pointing on the boundary of its domain $\mathcal{D} \subseteq \mathcal{F}$, the governor stays inside the free space interior $\mathring{\mathcal{F}}$ for all future times.

Since the reference planner r_{x^*} is Lipschitz continuous over its domain \mathcal{D} with a unique asymptotically stable point at x^* , by the converse Lyapunov theorem [21], there exists a smooth Lyapunov function $U: \mathbb{R}^d \rightarrow \mathbb{R}$ for r_{x^*} such that $\nabla U(y) \cdot r_{x^*}(y) < 0$ for all $y \in \mathcal{D} \setminus \{x^*\}$ [15]. Since the governor motion is determined by a nonnegatively scaled version of the reference planner in (12c), the governor dynamics in (13b) satisfies $\nabla U(y) \cdot y^{(1)} \leq 0$. Note that $\nabla U(y) \cdot y^{(1)} = 0$ only for $y = x^*$ and $\sigma(x, y) = 0$. If $\sigma(x, y)$ is zero, then it becomes positive in finite time since the closed-loop robot dynamics in (13a) is globally asymptotically stable at the governor position y contained in the free space interior $\mathring{\mathcal{F}}$. Hence, it follows from LaSalle's invariance principle [21] that both the robot and the governor asymptotically converge to zero motion at the goal x^* . ■

Hence, one can conclude from Propositions 1 & 2 that:

Theorem 1: (Safe & Stable Robot-Governor Navigation) Starting from any safe robot state $x \in (\mathbb{R}^d)^n$ and governor position $y \in \mathcal{D}$ with strictly positive safety level $\sigma(x, y) > 0$, the robot-governor dynamics in (13) asymptotically brings the n th-order robot and the first-order governor to the global goal x^* according to the first-order reference motion plan r_{x^*} with no collisions along the way.

IV. FROM LOW TO HIGH ORDER MOTION PLANNERS: EXAMPLE PLANNING & CONTROL ELEMENTS

In this section, we give design examples for low-level control and high-level planning elements that can be used in our motion planning framework. As low-level control elements, we provide an example choice of a standard feedback motion control policy for the n th-order robot dynamics, and present two example motion range prediction methods associated with that controller. As high-level planning element, we briefly describe an existing first-order path pursuit reference planner for safe path tracking in cluttered environment [15]. In the rest of the letter, the proofs are omitted due to space limitation and included in the technical report [20].

A. PhD Feedback Motion Control

A classical control approach for linear dynamical systems is negative error feedback [21]. Accordingly, we consider a simple negative error feedback policy to bring the n th-order robot model¹ to zero motion at any goal position.

Definition 2: (PhD Motion Control) For the n th-order robot dynamics, the *proportional higher-order derivative (PhD) control* that globally asymptotically stabilizes the robot state $x \in (\mathbb{R}^d)^n$ at any given goal (e.g., governor) position $y \in \mathbb{R}^d$ at zero

motion is defined as

$$\dot{x}^{(n)} = - \sum_{i=0}^{n-1} \kappa_i x^{(i)} + \kappa_0 y, \quad (18)$$

where the constant scalar control gains $\kappa_0, \dots, \kappa_{n-1} \in \mathbb{R}$ ensure that the characteristic polynomial $p(\lambda) = \lambda^n + \sum_{i=0}^{n-1} \kappa_i \lambda$ has roots with negative real parts.

To leverage tools from linear system theory, it is convenient to represent the PhD motion control in (18) as a first-order higher-dimensional dynamical system in the state space as

$$\dot{x} = (K \otimes I_{d \times d})(x - y), \quad (19)$$

where $K \in \mathbb{R}^{n \times n}$ is the (Hurtwitz) companion matrix associated with the control gains $\kappa_0, \dots, \kappa_{n-1}$, and $I_{d \times d}$ is the $d \times d$ identity matrix, and \otimes denotes the Kronecker product. Here, $x = (x^{(0)}, x^{(1)}, \dots, x^{(n-1)}) \in \mathbb{R}^{nd}$ and $y = (y, 0, \dots, 0) \in \mathbb{R}^{nd}$, respectively, denote the robot's state and the zero-motion goal (e.g., governor) state.

In practice, one often avoids using underdamped PD control for second-order robotic systems to prevent oscillatory robot motion. The notion of non-underdamped second-order systems can be extended to the PhD control of high-order dynamical systems as *non-overshooting* PhD control.

Definition 3 ([22]): (Non-overshooting PhD Control) A PhD feedback motion control of the form (18) is said to be *non-overshooting* if the control gains $\kappa_0, \dots, \kappa_{n-1}$ result in real negative characteristic polynomial roots, i.e., all eigenvalues of the associated companion matrix K are real and negative.

B. Convex Motion Range Predictions for PhD Control

We now present two motion range prediction methods based on Lyapunov ellipsoids and Vandermonde simplexes that provide a convex bound on the robot's motion trajectory under the PhD motion control [22].

1) *Lyapunov Motion Range Prediction for PhD Control:* Using the state-space form of the closed-loop robot dynamics under PhD control in (19), one can construct a quadratic Lyapunov function⁴ relative to the goal position y as

$$V_P(x) = (x - y)^T P (x - y) = \|x - y\|_P^2, \quad (20)$$

parametrized with a symmetric positive definite matrix⁵ $P \in S_{++}^n$ that uniquely satisfies the Lyapunov equation

$$(K \otimes I_{d \times d})^T P + P(K \otimes I_{d \times d}) + D^T D = 0, \quad (21)$$

for some matrix $D \in \mathbb{R}^{m \times n}$ such that $(K \otimes I_{d \times d}, D)$ is observable [21]. Since the time rate of change of the Lyapunov function satisfies

$$\begin{aligned} \frac{d}{dt} V_P(x) &= (x - y)^T \left((K \otimes I_{d \times d})^T P + P(K \otimes I_{d \times d}) \right) (x - y), \\ &= - \|D(x - y)\|^2 \leq 0, \end{aligned} \quad (22)$$

it follows from LaSalle's invariance principle [21] that the robot state $x(t)$, starting from any initial state $x(0) \in \mathbb{R}^{nd}$ is contained

⁴Here, $\|x\|_P$ denotes the weighted Euclidean norm associated with a positive definite matrix $P \in S_{++}^n$.

⁵ S_{++}^n denotes the set of $n \times n$ symmetric positive definite matrices.

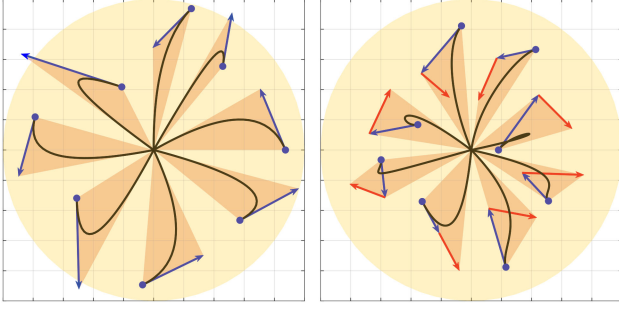


Fig. 2. Lyapunov (yellow circle) and Vandermonde (orange polygons) motion range prediction of PhD motion control towards the origin for (left) second- and (right) third-order robot dynamics. Here, the characteristic polynomial roots are uniformly spaced over $[-2, -1]$ and $\mathbf{D} = \mathbf{I}_{nd \times nd}$.

in the Lyapunov ellipsoid $\mathcal{E}(\mathbf{y}, \mathbf{P}^{-1}, \|\mathbf{x}(0) - \mathbf{y}\|_{\mathbf{P}})$,

$$\mathbf{x}(t) \in \mathcal{E}(\mathbf{y}, \mathbf{P}^{-1}, \|\mathbf{x}(0) - \mathbf{y}\|_{\mathbf{P}}) \quad \forall t \geq 0, \quad (23)$$

where $\mathcal{E}(\mathbf{c}, \mathbf{\Sigma}, \rho) := \{\mathbf{c} + \rho \mathbf{\Sigma}^{\frac{1}{2}} \mathbf{u} \mid \mathbf{u} \in \mathbb{R}^n, \|\mathbf{u}\| \leq 1\}$ is the ellipsoid centered at $\mathbf{c} \in \mathbb{R}^n$ and associated with a positive semidefinite matrix $\mathbf{\Sigma} \in \mathbb{S}_+^n$ and a nonnegative scalar $\rho \geq 0$, and $\mathbf{\Sigma}^{\frac{1}{2}}$ is a square root⁶ of $\mathbf{\Sigma}$ that satisfies $\mathbf{\Sigma}^{\frac{1}{2}} (\mathbf{\Sigma}^{\frac{1}{2}})^T = \mathbf{\Sigma}$.

To handle spatial safety constraints, one can bound the robot motion using an orthogonal projection of Lyapunov ellipsoids onto the position coordinates, as shown in Fig. 2.

Proposition 3: (Lyapunov Ellipsoids for PhD Control) Let $V_{\mathbf{P}}(\mathbf{x}) = (\mathbf{x} - \mathbf{y})^T \mathbf{P}(\mathbf{x} - \mathbf{y})$ be a quadratic Lyapunov function, parameterized with a positive definite matrix $\mathbf{P} \in \mathbb{S}_{++}^n$, for the stable closed-loop n th-order robot dynamics $\dot{\mathbf{x}} = (\mathbf{K} \otimes \mathbf{I}_{d \times d})(\mathbf{x} - \mathbf{y})$ under the PhD control in (19).

The robot motion trajectory $\mathbf{x}(t)$, starting at $t = 0$ from any initial state $\mathbf{x}(0) \in \mathbb{R}^{nd}$ towards a given goal position $\mathbf{y} \in \mathbb{R}^d$ is bounded by the projected Lyapunov ellipsoid as

$$\mathbf{x}(t) \in \mathcal{E}(\mathbf{y}, \mathbf{I}_{nd \times d}^T \mathbf{P}^{-1} \mathbf{I}_{nd \times d}, \|\mathbf{x}(0) - \mathbf{y}\|_{\mathbf{P}}) \quad \forall t \geq 0. \quad (24)$$

As expected, the projected Lyapunov ellipsoids simply satisfy the boundedness requirement in (7).

Proposition 4: (Bounding Ball of Lyapunov Ellipsoids) The projected Lyapunov ellipsoid $\mathcal{E}(\mathbf{y}, \mathbf{I}_{nd \times d}^T \mathbf{P}^{-1} \mathbf{I}_{nd \times d}, \|\mathbf{x} - \mathbf{y}\|_{\mathbf{P}})$ in (24) is bounded by the Euclidean ball centered at \mathbf{y} with radius $\eta \|\mathbf{x} - \mathbf{y}\|$, i.e.,

$$\mathcal{E}(\mathbf{y}, \mathbf{I}_{nd \times d}^T \mathbf{P}^{-1} \mathbf{I}_{nd \times d}, \|\mathbf{x} - \mathbf{y}\|_{\mathbf{P}}) \subseteq \mathcal{B}(\mathbf{y}, \eta \|\mathbf{x} - \mathbf{y}\|), \quad (25)$$

where $\eta = \|\mathbf{I}_{nd \times d}^T \mathbf{P}^{-1} \mathbf{I}_{nd \times d}\|^{\frac{1}{2}} \|\mathbf{P}\|^{\frac{1}{2}}$.

Projected Lyapunov ellipsoids also guarantee that the safety assessment in (10) is Lipschitz continuous.

Proposition 5: (Lipschitz Continuous Safety Assessment via Lyapunov Ellipsoids) The projected Lyapunov ellipsoids induce a Lipschitz continuous safety level measure $\sigma(\mathbf{x}, \mathbf{y}) = \text{dist}(\mathcal{E}(\mathbf{y}, \mathbf{I}_{nd \times d}^T \mathbf{P}^{-1} \mathbf{I}_{nd \times d}, \|\mathbf{x} - \mathbf{y}\|_{\mathbf{P}}), \partial \mathcal{F})$.

⁶One can compute the unique symmetric positive-definite square-root of $\mathbf{\Sigma}$ as $\mathbf{\Sigma}^{\frac{1}{2}} = \mathbf{V} \text{diag}(\sqrt{\sigma_1}, \sqrt{\sigma_2}, \dots, \sqrt{\sigma_n}) \mathbf{V}^T$ using the singular value decomposition $\mathbf{\Sigma} = \mathbf{V} \text{diag}(\sigma_1, \sigma_2, \dots, \sigma_n) \mathbf{V}^T$, where $\text{diag}(\lambda_1, \lambda_2, \dots, \lambda_n)$ is the diagonal matrix with elements $\lambda_1, \lambda_2, \dots, \lambda_n$.

2) **Vandermonde Motion Range Prediction:** As an alternative to Lyapunov ellipsoids, we consider Vandermonde simplexes [22] for bounding robot motion under non-overshooting PhD motion control (Definition 3). As seen in Fig. 2, Vandermonde simplexes offer significantly more accurate and less conservative motion range prediction compared to Lyapunov ellipsoids because Vandermonde simplexes have a stronger dependency on control parameters and robot state.

Proposition 6 ([22]): (Vandermonde Simplexes for PhD Control) Consider a non-overshooting PhD feedback motion controller for the n th-order robot model with control gains $\kappa_0, \dots, \kappa_{n-1} \in \mathbb{R}$ that are associated with real negative characteristic polynomial roots $\lambda_1, \dots, \lambda_n \in \mathbb{R}_{<0}$.

The robot trajectory $\mathbf{x}(t)$, starting at $t = 0$ from the initial state $\mathbf{x}(0) = (\mathbf{x}_0^{(0)}, \mathbf{x}_0^{(1)}, \dots, \mathbf{x}_0^{(n-1)})$ towards a given goal position $\mathbf{y} \in \mathbb{R}^d$, is contained in the Vandermonde simplex $\mathcal{V}_{\mathbf{y}}(\mathbf{x}(0))$

$$\mathbf{x}(t) \in \mathcal{V}_{\mathbf{y}}(\mathbf{x}(0))$$

$$:= \text{conv} \left(\mathbf{y}, \mathbf{x}_0^{(0)}, \mathbf{x}_0^{(0)} + \frac{\hat{\kappa}_1}{\hat{\kappa}_0} \mathbf{x}_0^{(1)}, \dots, \sum_{i=0}^{n-1} \frac{\hat{\kappa}_i}{\hat{\kappa}_0} \mathbf{x}_0^{(i)} \right), \quad (26)$$

for all $t \geq 0$, where conv denotes the convex hull operator, and the positive coefficients $\hat{\kappa}_0, \dots, \hat{\kappa}_{n-1}$ uniquely satisfy

$$\prod_{\substack{\lambda_i \in \{\lambda_1, \dots, \lambda_n\} \\ \lambda_i \neq \max(\lambda_1, \dots, \lambda_n)}} (\lambda - \lambda_i) = \sum_{i=0}^{n-1} \hat{\kappa}_i \lambda^i. \quad (27)$$

An important property of Vandermonde simplexes is that $\mathcal{V}_{\mathbf{y}}(\mathbf{x})$ is a linear transformation of the standard n -simplex $\Delta_n := \{(s_0, \dots, s_n) \in \mathbb{R}^{n+1} \mid \sum_{i=0}^n s_i = 1, s_i \geq 0 \forall i\}$, i.e.,

$$\mathcal{V}_{\mathbf{y}}(\mathbf{x}) = \{\mathbf{X} \mathbf{s} \mid \mathbf{s} \in \Delta_n\}, \quad (28)$$

where the transformation matrix \mathbf{X} is defined as a linear function of the robot state \mathbf{x} and the goal position \mathbf{y} as

$$\mathbf{X} = \left[\mathbf{y}, \mathbf{x}^{(0)}, \dots, \sum_{i=0}^{n-1} \frac{\hat{\kappa}_i}{\hat{\kappa}_0} \mathbf{x}^{(i)} \right] \in \mathbb{R}^{d \times (n+1)}. \quad (29)$$

As a motion range prediction method, Vandermonde simplexes satisfy the boundedness requirement in (7).

Proposition 7: (Bounding Ball of Vandermonde Simplexes) A Vandermonde simplex can be bounded by a Euclidean ball as

$$\mathcal{V}_{\mathbf{y}}(\mathbf{x}) \subseteq \mathcal{B}(\mathbf{y}, \eta \|\mathbf{x} - \mathbf{y}\|), \quad (30)$$

where $\eta = \sqrt{n \frac{\max(\hat{\kappa}_0, \dots, \hat{\kappa}_{n-1})}{\hat{\kappa}_0}}$.

Vandermonde simplexes also yield a Lipschitz continuous safety assessment as described in (10).

Proposition 8: (Lipschitz Continuous Safety Assessment via Vandermonde Simplexes) The Vandermonde-simplex-based safety level measure $\sigma(\mathbf{x}, \mathbf{y}) = \text{dist}(\mathcal{V}_{\mathbf{y}}(\mathbf{x}), \partial \mathcal{F})$ is Lipschitz continuous.

C. Path Pursuit Reference Planner

As a reference motion planner, we consider the “move-to-projected-path-goal” navigation policy in [15] that constructs a first-order vector field around a given navigation path based on a safe pure pursuit path tracking approach [23].

Let $\mathcal{P} : [0, 1] \rightarrow \hat{\mathcal{F}}$ be a continuous navigation path inside the free space interior $\hat{\mathcal{F}}$, either generated by a standard path

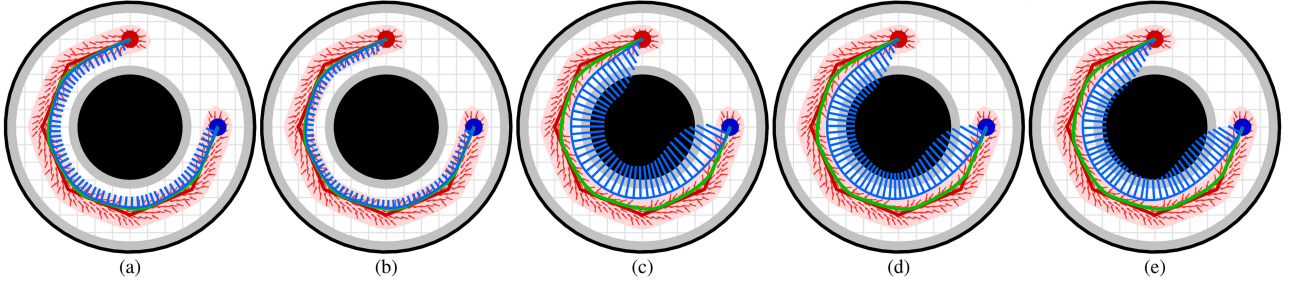


Fig. 3. Safe and smooth robot navigation over a circular corridor using a first-order path pursuit reference planner. The reference vector field (red arrows) are constructed based on a piecewise linear reference path (red line) starting at the blue circle and ending at the red circle. Workspace and configuration space obstacles are represented in black and gray, respectively. Robot motion is illustrated by blue lines, where blue bars indicate robot speed, and governor motion is represented by green lines. Robot motion range is predicted using Lyapunov ellipsoids in (a), (b) and Vandermonde simplexes in (c)–(e). Robot is assumed to have the second-order dynamics in (a), (c), the third-order dynamics in (b), (d), and the forth-order dynamics in (e).

planner [5] or determined by the user, that connects the start point $\mathcal{P}(0)$ to the end point $\mathcal{P}(1) = x^*$. Accordingly, the first-order “move-to-projected-path-goal” law (a.k.a. path pursuit reference planner) $r_{\mathcal{P}} : \mathcal{D}_{\mathcal{P}} \rightarrow \mathbb{R}^d$ is defined over its positively invariant domain $\mathcal{D}_{\mathcal{P}}$ [15],

$$\mathcal{D}_{\mathcal{P}} := \{q \in \mathcal{F} \mid \text{dist}(q, \mathcal{P}) \leq \text{dist}(q, \partial\mathcal{F})\}, \quad (31)$$

as

$$y^{(0)} = r_{\mathcal{P}}(y) = -\kappa_{\mathcal{P}}(y - \mathcal{P}^*(y)), \quad (32)$$

where $\kappa_{\mathcal{P}} > 0$ is a constant gain and the “projected path goal,” denoted by $\mathcal{P}^*(y)$, is determined as

$$\mathcal{P}^*(y) := \mathcal{P}\left(\max\left(\{\alpha \in [0,1] \mid \mathcal{P}(\alpha) \in B(y, \text{dist}(y, \partial\mathcal{F}))\}\right)\right). \quad (33)$$

By construction, for piecewise continuously differentiable navigation paths, the path pursuit planner $r_{\mathcal{P}}$ in (32) is locally Lipschitz continuous and inward pointing on its domain boundary $\partial\mathcal{D}_{\mathcal{P}}$, and it is asymptotically stable at $\mathcal{P}(1) = x^*$ whose domain of attraction includes the domain $\mathcal{D}_{\mathcal{P}}$ [15].

V. FROM LOW TO HIGH ORDER MOTION PLANNERS: NUMERICAL SIMULATIONS

In this section, we provide numerical simulations⁷ to demonstrate smooth extensions of the first-order path pursuit planner for safe navigation of high-order robots around obstacles, using the PhD motion control and safety assessment based on Lyapunov ellipsoids and Vandermonde simplexes. In particular, we systematically investigate the role of motion range prediction and system order on governed robot motion.

A. Safe Navigation in a Circular Corridor

As a first example, we consider safe navigation in a circular narrow corridor since smooth motion planning and control in such tight spaces is a well-known challenge for high-order robots [16]. In Fig. 3, we illustrate the resulting robot trajectories and speeds for the second-, third-, and fourth-order

⁷For all simulations, we set the path pursuit planner gain $\kappa_{\mathcal{P}} = 1$, the governor gain $\kappa_g = 4$ and the characteristic polynomial roots of the PhD control are uniformly spaced over $[-2, -1]$. All simulations are obtained by numerically solving the associated robot-governor dynamics using the `ode45` function of MATLAB. Please see the accompanying video for the animated robot-governor motion.

robot dynamics, where the safety of robot motion relative to the governor is monitored using Lyapunov ellipsoids and Vandermonde simplexes. As expected, the robot can reach to the desired destination of the path pursuit reference planner irrespective of the order of robot dynamics and motion range prediction, but the resulting robot motion significantly differs in terms of robot speed and so travel time. As seen in Fig. 2, Lyapunov ellipsoids are more conservative in estimating robot motion range, because Vandermonde simplexes has stronger dependency on robot’s state and control parameters (see Propositions 3 and 6). As a result, Vandermonde simplexes always yield faster robot motion in Fig. 3. We observe that the robot is more cautious about sideways collisions with corridor walls when its motion range is predicted by Lyapunov ellipsoids. Also, since motion range prediction is naturally more conservative for higher-order robot dynamics, we see robot motion gets slower with increasing system order.

B. Safe Navigation in a Cluttered Environment

To demonstrate how motion prediction plays a critical role in adapting robot motion around complex obstacles, we consider safe robot navigation in a cluttered environment, illustrated in Fig. 4. In such a environment, one might naturally expect that the robot slows down while making a turn around obstacles and speeds up if there is a large opening in front of the robot. This, of course, depends on many design parameters in planning and control. We observe in our numerical studies that motion prediction is a key factor. Conservative motion prediction such as Lyapunov ellipsoids often has a tendency to slow down robot motion because the predicted robot motion cannot be accurately related to the environment. As seen in Fig. 4, Lyapunov motion range prediction is limited in adapting robot motion around obstacles, whereas Vandermonde simplexes allow the robot to leverage available space for faster navigation without compromising safety since Vandermonde simplexes are able to capture robot motion more accurately. Therefore, accurate motion prediction is crucial for generating adaptive agile highly dynamic robot motion around complex (potentially dynamic) obstacles. Finally, it is useful to note that faster robot trajectories significantly deviate from the reference (governor) trajectories because a reference plan in our motion planning framework is considered as a high-level flexible navigation plan towards the destination point as opposed to precise trajectory tracking control [24].

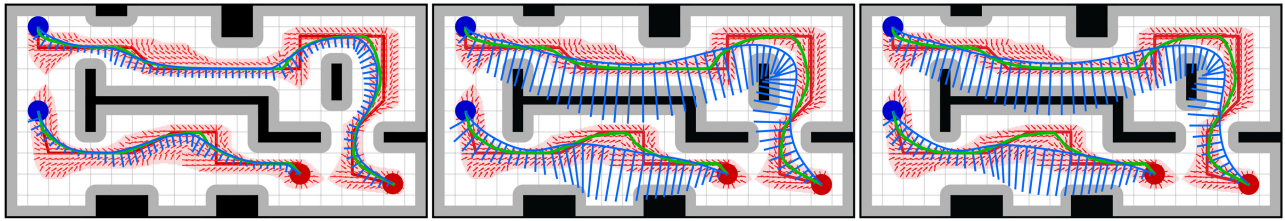


Fig. 4. Safe and smooth robot navigation in a cluttered environment using a first-order path pursuit reference planner (red arrows) that is constructed based on a piecewise linear reference path (red line). Robot motion (blue lines) for second-order (left, middle) and third-order (right) dynamics is constantly predicted relative to governor motion (green lines) using (left) Lyapunov ellipsoids and (middle, right) Vandermonde simplexes, where the robot speed is indicated by blue bars.

VI. CONCLUSION

In this letter, we introduce a provably correct feedback motion planning framework that extends safe navigation properties of simple first-order reference motion planners to high-order robot dynamics using reference governors and safety assessment of predicted robot motion. Our approach separates high-level planning and low-level control challenges to address them individually using standard tools from the motion planning and control literature. We establish a novel bidirectional interface between planning and control using reference governors and motion prediction. In addition to specifying generic motion planning elements, we provide example designs for motion control, prediction, and reference planning. We conclude that accurate motion prediction plays a key role in closing the gap between high-level planning and low-level control for generating agile robot motion. In numerical simulations, we demonstrate the effectiveness of the proposed motion planning framework.

Work now in progress focuses on extending our motion planning framework to nonholonomically constrained robots such as differential drive vehicles and drones by designing new nonholonomic motion prediction algorithms. Another promising research direction is sensor-based safe robot navigation in dynamic and unknown environments [25].

REFERENCES

- [1] B. Donald, P. Xavier, J. Canny, and J. Reif, "Kinodynamic motion planning," *J. ACM*, vol. 40, no. 5, pp. 1048–1066, 1993.
- [2] S. M. LaValle and J. J. Kuffner Jr., "Randomized kinodynamic planning," *Int. J. Robot. Res.*, vol. 20, no. 5, pp. 378–400, 2001.
- [3] T. Fraichard and H. Asama, "Inevitable collision states - A step towards safer robots?," *Adv. Robot.*, vol. 18, no. 10, pp. 1001–1024, 2004.
- [4] A. Ravankar, A. A. Ravankar, Y. Kobayashi, Y. Hoshino, and C.-C. Peng, "Path smoothing techniques in robot navigation: State-of-the-art, current and future challenges," *Sensors*, vol. 18, no. 9, 2018, Art. no. 3170.
- [5] S. M. LaValle, *Planning Algorithms*. Cambridge, U.K.: Cambridge Univ. Press, 2006.
- [6] C. Richter, A. Bry, and N. Roy, "Polynomial trajectory planning for aggressive quadrotor flight in dense indoor environments," in *Robotics Research*, Berlin, Germany: Springer, 2016, pp. 649–666.
- [7] W. Ding, W. Gao, K. Wang, and S. Shen, "An efficient b-spline-based kinodynamic replanning framework for quadrotors," *IEEE Trans. Robot.*, vol. 35, no. 6, pp. 1287–1306, Dec. 2019.
- [8] J. Tordesillas, B. T. Lopez, M. Everett, and J. P. How, "FASTER: Fast and safe trajectory planner for navigation in unknown environments," *IEEE Trans. Robot.*, vol. 38, no. 2, pp. 922–938, Apr. 2022.
- [9] S. Liu *et al.*, "Planning dynamically feasible trajectories for quadrotors using safe flight corridors in 3-D complex environments," *IEEE Robot. Autom. Lett.*, vol. 2, no. 3, pp. 1688–1695, Jul. 2017.
- [10] W. Ding, L. Zhang, J. Chen, and S. Shen, "Safe trajectory generation for complex urban environments using spatio-temporal semantic corridor," *IEEE Robot. Autom. Lett.*, vol. 4, no. 3, pp. 2997–3004, Jul. 2019.
- [11] M. Chen *et al.*, "FasTrack: A modular framework for real-time motion planning and guaranteed safe tracking," *IEEE Trans. Autom. Control*, vol. 66, no. 12, pp. 5861–5876, Dec. 2021.
- [12] A. Bemporad, "Reference governor for constrained nonlinear systems," *IEEE Trans. Autom. Control*, vol. 43, no. 3, pp. 415–419, Mar. 1998.
- [13] E. Gilbert and I. Kolmanovsky, "Nonlinear tracking control in the presence of state and control constraints: A generalized reference governor," *Automatica*, vol. 38, no. 12, pp. 2063–2073, 2002.
- [14] E. Garone and M. M. Nicotra, "Explicit reference governor for constrained nonlinear systems," *IEEE Trans. Autom. Control*, vol. 61, no. 5, pp. 1379–1384, May 2016.
- [15] Ö. Arslan and D. E. Koditschek, "Smooth extensions of feedback motion planners via reference governors," in *Proc. IEEE Int. Conf. Robot. Automat.*, 2017, pp. 4414–4421.
- [16] Z. Li, Ö. Arslan, and N. Atanasov, "Fast and safe path-following control using a state-dependent directional metric," in *Proc. IEEE Int. Conf. Robot. Automat.*, 2020, pp. 6176–6182.
- [17] Z. Li, T. Duong, and N. Atanasov, "Safe robot navigation in cluttered environments using invariant ellipsoids and a reference governor," 2020, arXiv: 2005.06694.
- [18] A. D. Luca, G. Oriolo, and M. Vendittelli, "Stabilization of the unicycle via dynamic feedback linearization," *IFAC Symp. Robot Control*, vol. 33, no. 27, pp. 687–692, 2000.
- [19] D. Zhou and M. Schwager, "Vector field following for quadrotors using differential flatness," in *Proc. IEEE Int. Conf. Robot. Autom.*, 2014, pp. 6567–6572.
- [20] A. İşleyen, N. van de Wouw, and Ömür Arslan, "From low to high order motion planners: Safe robot navigation using motion prediction and reference governor (Technical Report)," 2022, arXiv:2202.12816.
- [21] H. K. Khalil, *Nonlinear Systems*. Hoboken, NJ, USA: Prentice Hall, 2001.
- [22] Ö. Arslan and A. İşleyen, "Simplicial trajectory bounds for linear companion systems via Vandermonde basis," 2022.
- [23] R. C. Coulter, "Implementation of the pure pursuit path tracking algorithm," Carnegie Mellon Univ., Robot. Inst., Pittsburgh PA, Tech. Rep., 1992.
- [24] S. Quinlan and O. Khatib, "Elastic bands: Connecting path planning and control," in *Proc. IEEE Int. Conf. Robot. Autom.*, 1993, pp. 802–807.
- [25] Ö. Arslan and D. E. Koditschek, "Sensor-based reactive navigation in unknown convex sphere worlds," *Int. J. Robot. Res.*, vol. 38, no. 2/3, pp. 196–223, 2019.