Fall 11-15-2018

Scalable Stochastic Reachability: Theory, Computation, and Control

Abraham Puthuvana Vinod
University of New Mexico - Main Campus

Abraham Puthuvana Vinod

Candidate

Electrical and Computer Engineering

Department

This dissertation is approved, and it is acceptable in quality and form for publication:

Approved by the Dissertation Committee:

Meeko Oishi , Chairperson

Chaouki Abdallah

Rafael Fierro

Lydia Tapia

Majeed Hayat

Scalable Stochastic Reachability: Theory, Computation, and Control

by

Abraham Puthuvana Vinod

B. Tech, Electrical Engineering, Indian Institute of Technology, Madras, 2014

M. Tech, Electrical Engineering, Indian Institute of Technology, Madras, 2014

DISSERTATION

Submitted in Partial Fulfillment of the
Requirements for the Degree of

Doctor of Philosophy
Engineering

The University of New Mexico
Albuquerque, New Mexico

December, 2018

Dedication

To my family

Acknowledgements

I am grateful to everyone around me for making my graduate school life meaningful and productive. Most importantly, I am grateful to my advisor, Meeko Oishi, for asking probing questions that led to this dissertation, for giving me the complete freedom to follow my intuitions to address these questions, and for being an excellent mentor and friend. I am also grateful to Lydia Tapia and Rafael Fierro for our collaborations that has enriched this dissertation with real-world applications. I am grateful to Chaouki Abdallah for giving me valuable feedback throughout my graduate school life. I am indebted to Majeed Hayat and Cristina Pereyra for their courses on probability theory and real analysis and their availability to discuss these vast topics, without which I could not have written this dissertation. I am also grateful to Behçet Açıkmeşe at the University of Washington and Tyler Summers at the University of Texas, Dallas for the fruitful collaborations.

In these past few years, I have had the good fortune to work with and befriend astounding colleagues. I am glad to have met and worked with Joseph, who has been a great collaborator and friend. I am proud of the toolbox we wrote together. I am also grateful to have met Kendra, Vignesh, Adam, Harini, Jessica, Pankaj, and Manish. On a personal note, I am glad to have the numerous friends I made in Albuquerque, who gave me more support and love than I could have ever hoped for.

Last but definitely not the least, I thank my family for their love, support, and encouragement throughout this journey. This work would have not been possible without my parents, my brother Paul, and the love of my life, Rosina. I dedicate this work to them.

Scalable Stochastic Reachability: Theory, Computation, and Control

by

Abraham Puthuvana Vinod

B. Tech, Electrical Engineering, Indian Institute of Technology, Madras, 2014

M. Tech, Electrical Engineering, Indian Institute of Technology, Madras, 2014

Ph.D., Electrical Engineering, University of New Mexico, 2018

Abstract

Guaranteeing safety and performance are crucial components in any control system, and particularly relevant in light of growing interest in reliable autonomy. In safety-critical applications like biomedical devices, spacecraft applications, and self-driving cars, the cost of failure can be severe. Verification provides these guarantees by characterizing the "good" initial states or configurations from which a state can be driven to remain within a collection of pre-specified safe sets, while respecting the system dynamics, bounds on control authority, and additive uncertainties. We also wish to design controllers to achieve this objective. This dissertation proposes novel theory and scalable algorithms for tractable solutions to the stochastic reachability problems.

We treat the verification problems as *backward stochastic reachability* problems, since we wish to ascertain a set of initial states (back in time) that satisfy the pre-specified safety constraints in future and determine the associated admissible con-

trollers. Current approaches to this problem are restricted to low dimensional systems since they rely on grids over the state space. In contrast, we have developed several approaches which avoid gridding, scale well with time and system dimension, and may be terminated at any time without compromising on the solution integrity (anytime algorithms). Additionally, our approaches enable controller synthesis that is open-loop or affine feedback. We design these algorithms by characterizing the sufficient conditions for the convexity and compactness of the sets. This enables verification of high-dimensional systems using convex optimization, stochastic programming, and Fourier transforms. We apply our methods to verification of spacecraft rendezvous, as well as to automated anesthesia delivery.

We also discuss a related problem of predicting the stochasticity of the state of the system at a future time, given an initial state — the problem of *forward stochastic reachability*. We address this problem using Fourier transforms and computational geometry. Using forward stochastic reachability, we also introduce *probabilistic occupancy functions* to reason about collision probability and characterize keep-out sets for probabilistic safety. The applications include autonomous surveillance and stochastic motion planning problems.

We present an open-source MATLAB toolbox, `SReachTools`, that implements all the algorithms presented in this dissertation in an easily-accessible code base.

Contents

List of Figures

List of Tables

Chapter 1

Introduction

1.1 Motivation

One of the major goals of control system theory is to design of safe and efficient autonomy. These safety and performance guarantees come to the forefront in safety-critical applications like biomedical devices, spacecraft applications, and self-driving cars where the cost of failure is severe. This goal is also seen in robotics and motion planning problems, where the system must remain within pre-specified operation parameters for continued operation. Reachability analysis provides a rigorous mathematical framework to address these verification problems [Gir05; MBT05; Aba+08; SL10]. In this dissertation, we propose novel theory and algorithms to perform scalable stochastic reachability analysis (reachability in presence of stochastic disturbance) and controller synthesis.

We discuss two motivating applications for the verification problems of interest in this thesis — the problems of *forward stochastic reachability* and *backward stochastic reachability*. Informally, forward stochastic reachability analysis characterizes the stochasticity of the state of a system at a future time in interest. In contrast, the backward stochastic reachability analysis determines the set of safe initial states and

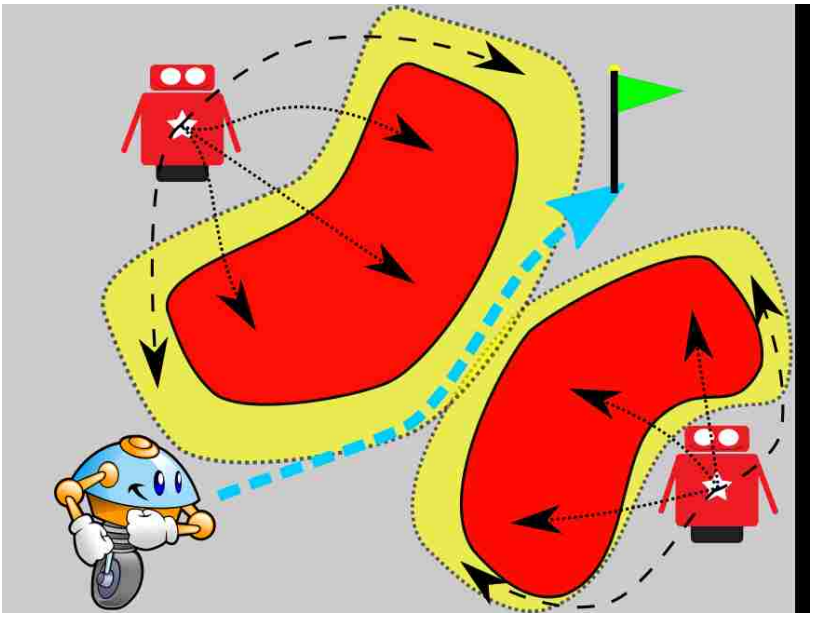


Figure 1.1: We wish to drive the blue robot towards a goal while avoiding the red obstacles which have stochastic dynamics. The red regions are the keep-out regions to achieve a moderate probabilistic safety, and the blue trajectory is a feasible probabilistically safe trajectory. On the other hand, avoiding the yellow regions guarantee a higher probabilistic safety, but the problem becomes infeasible.

their associated controllers that satisfies a given safety specification within a time horizon.

Figure 1.1 describes a stochastic motion planning problem which requires forward stochastic reachability. To ensure the safe operation of the blue robot, we require the autonomous controller to design safe trajectories while accounting for the dynamics of the blue robot, its actuation limits, and constraints arising from the environment. In this problem, we have multiple obstacles (shown in red) that have complex geometries and stochastic dynamics. Apart from existing constraints on the trajectory, the autonomous controller must ensure that the *probabilistic safety* of the blue robot is above a specified threshold. This is enforced by requiring the collision probability, the probability that the blue robot hits one of the red obstacles at some time in future when it is at a particular location, is below a corresponding threshold. Such stochastic motion planning problems are ubiquitous in robotics [Mal+17; HVO17] and autonomous

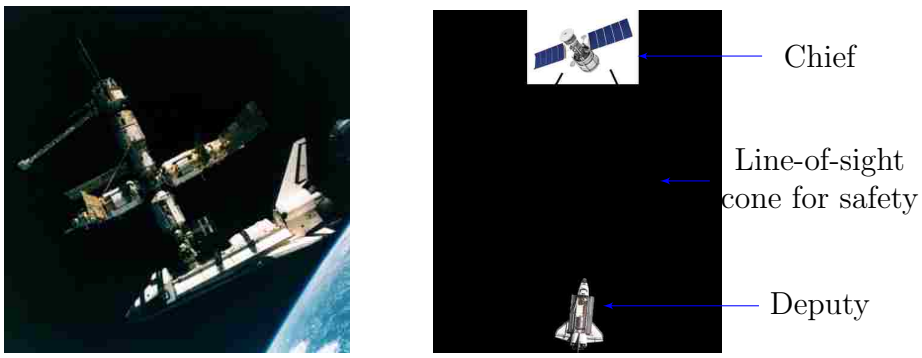


Figure 1.2: Space applications require safe spacecraft rendezvous for maintainance and resupply missions. For safe operation, we require the autonomous controller driving the deputy spacecraft to achieve rendezvous with the chief spacecraft, while staying within a line-of-sight cone. Picture courtesy: NASA

transportation [Goo18; Sum+11]. Using forward stochastic reachability techniques, we can characterize the stochasticity associated with the states of the obstacles at future time instants. We also define the notion of probabilistic occupancy functions to reason about the collision probability. In Figure 1.1, the red regions correspond to the states which the red robots are highly likely to occupy resulting in high collision probability, whereas the yellow regions include the states which the red obstacles are less likely to occupy. Using probabilistic occupancy function, we can construct a collection of time-stamped keep-out regions that the blue robot must respect to guarantee a desired level of probabilistic safety. These keep-out regions can be given directly to existing deterministic motion planners to plan trajectories that meet the specified probabilistic safety requirements. By characterizing sufficient conditions for their convexity and compactness, we decrease the computational effort necessary to plan these probabilistically safe trajectories. As seen in Figure 1.1, it is possible that the motion planning problem is infeasible for very high safety probabilities.

Figure 1.2 describes a more realistic problem of a spacecraft rendezvous problem, whose solution requires backward stochastic reachability analysis. In this problem, we wish to autonomously maneuver a spacecraft, known as the "deputy", to rendezvous with another spacecraft, known as the "chief". The relative dynamics of the deputy

spacecraft with respect to the chief, when both of these spacecrafts are in the same circular orbit, are linear (Clohessy-Wiltshire-Hill dynamics) with additive stochastic disturbance arising from the model uncertainties. The safety specification arises from the need for accurate sensing. Here, the deputy must remain in a line-of-sight cone and approach the origin, the location of the chief in the relative coordinate frame. The automation on the deputy must also satisfy the actuator limits. To ensure safe operation of this safety-critical and expensive application, we must identify the set of "good" initial locations from which this safety specification can be met with a probability of success above a desired threshold, while respecting the dynamics and limits on actuation. Finally, we also require synthesis of admissible controllers for the automation on the deputy spacecraft.

This thesis covers theoretical foundations and scalable algorithms to address both of these problems. Current state-of-the-art approaches rely on grids over the state space, limiting its application to low dimensional system. We use convex optimization, stochastic programming, Fourier transforms, and computational geometry to propose scalable and grid-free algorithms that outperform the current state-of-the-art approaches. We establish sufficient conditions for convexity and compactness of the sets of interest, exploit these properties for polytopic approximations, and leverage standard solvers to produce non-conservative probabilistic safety guarantees and controller synthesis for these problems. We also present an open-source MATLAB toolbox, `SReachTools`, that implements the proposed algorithms in a user-friendly and extensible codebase.

1.2 Uncertainty in Control

Uncertainty is a common element in control problems. It may arise from various sources — modeling limitations, disturbance effects (like weather), actions of

human(s)-in-the-control-loop, and sensor noise to list a few. In control theory, uncertainties are typically dealt either in a *robust sense* or in a *stochastic sense*. In the robust approach, the controllers are designed to handle the worst-case uncertainty realization. This approach requires only bounds on the uncertainty and provides absolute guarantees of safety. On the other hand, when stochastic information about the uncertainty is available, we pursue probabilistic guarantees where the system can be determined to be safe above a specified likelihood. The stochasticity information of the uncertainty permits assessment of safety in a gradated fashion. Specifically, it provides a degree of safety (a probabilistic guarantee of safety) instead of a binary safe or not-safe assessment, as in the robust case. In problems that permit a small margin of safety, say probability of collision must not be above 0.001, the robust approach will provide overly conservative results, when compared to the stochastic approach.

For example, in the stochastic motion planning problem discussed in Figure 1.1, the robust approach will require the blue robot to avoid a collection of sets that includes all possible configurations of the red obstacles that may cause collision (accounting for the worst-case possibilities). This set is guaranteed to contain the yellow sets, since the yellow sets exclude the highly unlikely configurations which the keep-out sets associated with the robust approach include. But as seen in this example, the motion planning problem is infeasible when using yellow keep-out sets. In other words, the associated very high safety probability requirement leads to an infeasible motion planning problem. Consequently, the robust approach will also lead to an infeasible problem, since it demands absolute safety. On the other hand, reducing the probabilistic safety requirement (red sets) can provide feasible trajectories. Thus, the stochastic description of the uncertainty permits a mathematical framework to tradeoff performance with safety, which is critical in real-world applications. Note that the reduced conservativeness in the stochastic motion planning problems is a natural result of the more detailed mathematical model.

1.3 Stochastic Reachability: Theory and Computation

Reachability analysis provides a mathematical framework to determine if a control system satisfies a given specification like safety. It also provides techniques for controller synthesis in order to satisfy the given specification. We will consider the problem of stochastic reachability, i.e., reachability analysis for systems with stochastic uncertainty.

Stochastic reachability analysis has been developed for the general class of discrete-time stochastic hybrid systems [Aba+07; Aba+08; SL10]. This generic class of system provides a rigorous mathematical framework to model systems with discrete and continous elements, with stochastic and controlled discrete and continuous state transitions. A dynamic programming solution was proposed to perform stochastic reachability analysis for these systems [Aba+07]. However, this approach suffers significant computational costs when used on high-dimensional systems, due to its reliance on a grid over the continuous state space. A significant portion of this thesis is dedicated to alleviating this lack of scalability by constructing theory and algorithms that can perform stochastic reachability analysis without relying on a grid. We will restrict our focus to nonlinear time-varying systems for the development of theory, and to linear time-varying systems for computational algorithms. These restrictions arise from the sufficient conditions we propose to attain convexity and compactness, the key enabler for scalable and grid-free stochastic reachability analysis (see Table 4.1).

1.3.1 Forward Stochastic Reachability

Forward stochastic reachability analysis of a discrete-time stochastic system characterizes two properties associated with the system state at a future time of interest:

1. its associated probability measure (the forward stochastic reach probability

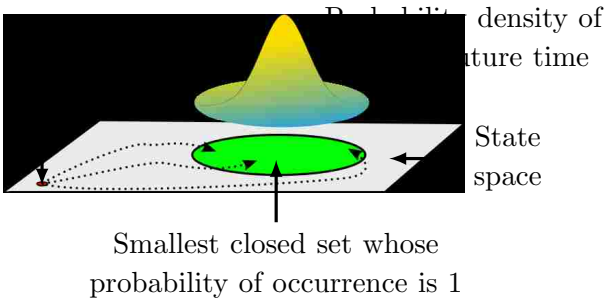


Figure 1.3: Forward stochastic reachability problem

measure) or density, and

2. its support, the smallest closed set that covers all the reachable states (the forward stochastic reach set).

These properties, illustrated in Figure 1.3, can help us answer questions of the form: *what is the likelihood that the system at a future time of interest will lie in a target set? Is there a non-zero likelihood of the state lying in a given collection of states at a future time of interest?*

As an illustrative example, consider a point mass system initialized at the origin with its velocities drawn from a a truncated Gaussian. Figure 1.4 shows the forward stochastic reach set and the probability measure (truncated Gaussian) after 7 time steps. As expected, the probability density and forward stochastic reach set spreads over larger areas of the state space over time, due to the propagation of the uncertainty through the point mass dynamics.

Note that the forward stochastic reach probability density and reach set have been analyzed separately in control theory literature for some special cases. For Gaussian-perturbed linear systems, the probability measure at a future time of interest can be obtained from the prediction steps of a Kalman filter [DCA94]. However, this approach fails to generalize for non-Gaussian disturbances, since it tracks only the first two moments of the state. Similarly, for LTI systems with bounded disturbances, established verification methods [Kva+15; KV06; Gir05] can be adapted to overapproximate the forward stochastic reachable set. However, these methods return a

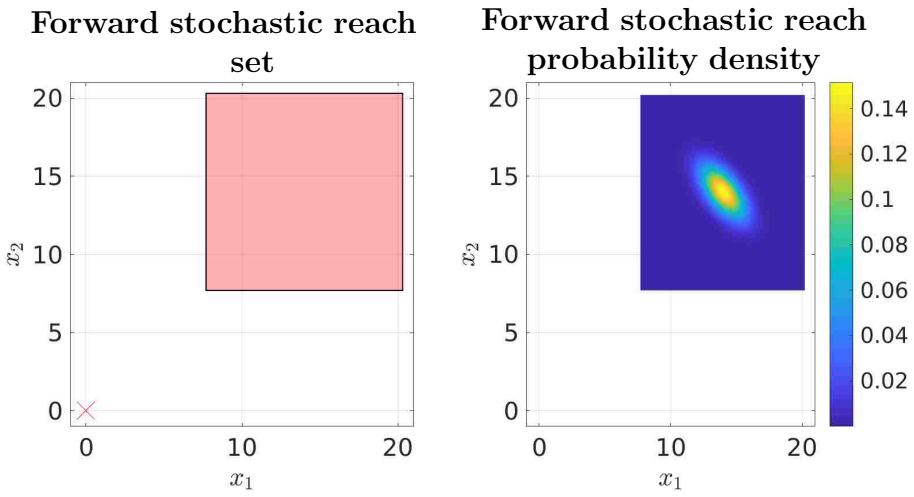


Figure 1.4: Forward stochastic reachability analysis for a discrete-time point mass system perturbed by a truncated Gaussian disturbance after 7 time steps, when initialized at the origin. The truncated Gaussian had a non-zero mean and non-identity covariance matrix.

trivial result with unbounded disturbances and do not address the forward stochastic reach probability measure, which provides the likelihood of reaching a given set of states. Alternatively, one can use grid-based dynamic programming approaches to compute these quantities, which do not scale well with system dimension [Aba+07].

In this thesis, we propose a scalable and grid-free approach to characterize all possible realizations of the state and the associated likelihoods using Fourier transforms and computational geometry. We also discuss sufficient conditions under which the forward stochastic reach probability measure or density are log-concave and the forward stochastic reach set is convex. We also apply forward stochastic reachability to the problem of stochastic motion planning (Figure 1.1) and the problem of stochastic target capture.

Application: Stochastic Motion Planning

In stochastic motion planning problems (Figure 1.1), we wish to navigate a controllable robot with known dynamics in a known environment that has rigid-body obstacles with known stochastic dynamics. Several approaches exist for obstacle avoidance

with static obstacles [KF11; LaV06; SFH02; Mao+17; BOW11], with growing research in stochastic obstacle avoidance [TBF05; LKH10; Mal+17; Chi+15; Chi+17]. See [Mal+17, Sec. 2] for a more detailed survey. However, this problem remains challenging to due to 1) the inherent non-convexity in the problem, and 2) lack of efficient techniques to predict collision probability and identify bad states in which collision is highly likely. Using forward stochastic reachability, we propose efficient techniques to address the second challenge. Specifically, we construct *probabilistic occupancy functions* to quantify the collision probability and characterize the keep-out sets (probabilistic occupied sets) to guarantee probabilistically safe trajectories. Probabilistic occupancy functions provide the probability with which a given state at a particular time of interest is occupied by one of the rigid-body obstacles. The superlevel sets of the probabilistic occupancy function provide the keep-out regions. We also propose sufficient conditions under which the keep-out regions are closed, bounded, compact, and convex (or a union of convex) sets. We then exploit these set-theoretical properties to propose computationally efficient, scalable, grid-free, and anytime algorithms to overapproximate the keep-out regions. Recall that anytime algorithms provide a valid solution, even if terminated early. To demonstrate the utility of these keep-out sets, we utilize existing successive convexification trajectory planners [MSA16; Mao+17] and receding horizon control to design probabilistically safe trajectories for a stochastic motion planning problem.

Application: Stochastic Target Capture

We can also utilize forward stochastic reachability in the problem of stochastic non-adversarial target capture which may arise, in e.g., the rescue of a lost first responder in a building on fire [KRS04], capture of a non-aggressive UAV in an urban environment [Gey08], or other non-antagonistic situations [Hol+09]. Informally, we wish to maximize the probability of capture of a non-adversarial target with stochastic

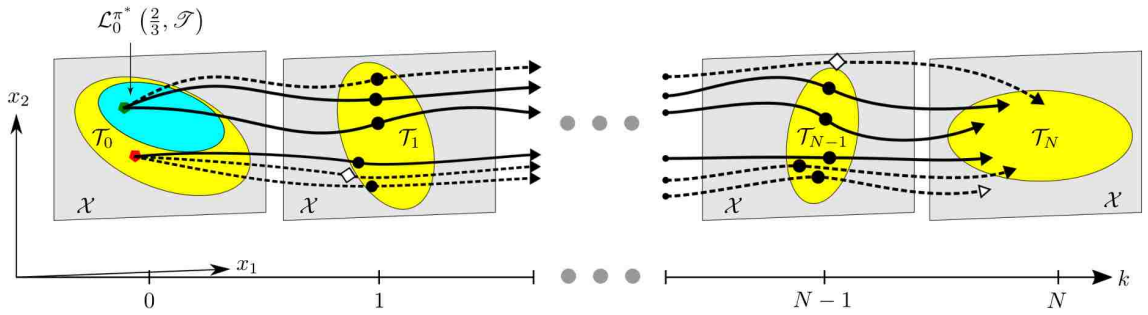


Figure 1.5: Stochastic reachability of a target tube characterizes the set $\mathcal{L}_0^{\pi^*}(\alpha, \mathscr{T})$ and an admissible controller π^*. For any initial state in $\mathcal{L}_0^{\pi^*}(\alpha, \mathscr{T})$, the state of the stochastic system stays within the target tube $\mathscr{T} = \{\mathcal{T}_k\}_{k=0}^N$, $\mathcal{T}_k \subseteq \mathcal{X}$, with a likelihood higher than α. This problem will be formulated in Section 4.3.

dynamics using a controllable pursuer while respecting the pursuer dynamics. While solutions for an adversarial target, based in a two-person, zero-sum differential game, can accommodate bounded disturbances with unknown stochasticity [MT00; TLS00; Tom+03; BFZ10; Hua+15], it will be conservative for a non-adversarial target, modeled using stochastic dynamics.

We seek scalable solutions that synthesize an optimal controller for the non-adversarial scenario, by exploiting the forward reachable set and probability measure for the target. Specifically, we use convex optimization to provide the exact probabilistic guarantee of success and the corresponding optimal controller. We experimentally validate our approach on a quadrotor testbed.

1.3.2 Backward Stochastic Reachability

We frame the problem of backward stochastic reachability as the problem of stochastic reachability of a target tube, motivated by the question: *what initial states of a stochastic dynamical system can be driven to stay within a target tube (a collection of time-stamped target sets) with a desired likelihood, while respecting the given bounds on control authority?* Figure 1.5 provides an illustration of this problem. The qualifier "backward" arises from the fact that we wish to acertain properties of initial system

states based on safety specifications defined for the future states.

The problem of stochastic reachability of target tube subsumes existing work on stochastic viability and terminal hitting-time stochastic reach-avoid problems [Aba+08; SL10]. Stochastic viability problems are concerned with maximizing the probability that system stays within a time-invariant safe set for a given time horizon (target tube with time-invariant safe sets) [Aba+08]. Terminal hitting-time stochastic reach-avoid problems are concerned with maximizing the probability that system stays within a time-invariant safe set within the time horizon and hits a (potentially different) target set at the time horizon [SL10].

Theory

The problem of backward stochastic reachability has received significant attention in the verification literature. The dynamic programming formulation for the stochastic reachability problem [Aba+07; Aba+08; SL10], casts the stochastic reachability problem as a discrete-time stochastic optimal control problem (via Markov decision process theory) [BS78]. It yields optimal value functions which map the states to their maximal reach probability. The superlevel sets of these functions, the *stochastic reach sets*, are the sets of "good" initial states, i.e., the set of initial states from which the system may be driven to stay within the target tube with a probability greater than a given threshold. For stochastic reach-avoid problems, sufficient conditions have been proposed for the well-posedness of the stochastic reach-avoid problem and the existence of an optimal Markov policy [Din+13; Kar+17; Yan18; VO17]. However, little is known about the sufficient conditions that guarantee convexity and compactness of stochastic reach sets, which we address in this thesis. Specifically, we propose sufficient conditions under which the optimal value functions are Borel measurable, upper semicontinuous, and log-concave, and the stochastic reach set is closed, bounded, compact, and convex. These theoretical results enable the construction of

tight polytopic underapproximations (inner-approximations), that allow us to design to scalable, grid-free, and anytime algorithms to verify high-dimensional systems.

Computation and controller synthesis

For the computation of the maximal reach probability, a numerical implementation of the dynamic programming has been proposed, which requires gridding the continuous state space [SB98; Aba+07]. The reliance on a grid over the state space translates to an exponentially increasing computational cost as the system dimension increases, making this approach intractable for system dimensions higher than three or four. Researchers have focused on alleviating the *curse of dimensionality* via approximate dynamic programming [KML16; Man+15], Gaussian mixtures [KML16], particle filters [Man+15; LOE13], convex chance-constrained optimization [LOE13], and semi-definite programming [Drz+16; Kar+17]. These techniques have been applied to systems that are at most 10-dimensional — far beyond the scope of what is possible with dynamic programming, but not scalable to larger problems.

With respect to controller synthesis, the dynamic programming provides an optimal Markov controller in the form of a look-up table for low-dimensional systems [SB98; Aba+07]. Almost all of the above-mentioned approximative approaches seek open-loop control laws for tractability, at the cost of significant conservativeness. Approximate dynamic programming techniques can synthesize closed-loop controllers by parameterizing the policy space [Man+15; Kar+17]. However, these approximation approaches do not guarantee an underapproximation of the safety probability. This limits its application to safety-critical applications, where we can not afford to be overconfident with the degree of safety.

In the past few decades, the controller synthesis problem in the stochastic optimal control problems has received a lot of attention. Specifically, techniques based on stochastic receding horizon control (also known as stochastic model predictive control)

have been developed that can synthesize (sub)optimal controllers to drive the system while satisfying the stochastic dynamics, probabilistic state and input constraints, and minimizing a cost function. See [Mes16; FGS16] for recent surveys on this problem. However, these techniques can not be used to compute the set of good initial states which is the main focus of this thesis.

For the case of non-stochastic uncertainty, several approaches based on robust optimal control exist to identify the backward reach sets. For continuous time systems, techniques using Hamilton-Jacobi formulation and level set techniques have been proposed [MBT05]. For discrete time systems, techniques using computational geometry are available for linear systems [BR71; Gir05; KV00; LGG09].

In this thesis, we present four novel approaches to evaluate the optimal safety probability and the associated optimal (open-loop or affine) controller using convex optimization, stochastic programming, and Fourier transforms. All of these approaches are grid-free and scale well with dimension when compared to the dynamic programming approach [Aba+07]. We also combine these point-based verification and controller synthesis techniques with a ray-shooting algorithm to underapproximate the sets in a scalable, grid-free, and anytime algorithm. This approach enables, for the first time, the verification of systems as high as 40-dimensions.

Applications

We will demonstrate applications for backward stochastic reachability in the spacecraft rendezvous problem (Figure 1.2) and the automated anesthesia delivery system [Aba+18]. Researchers have also used stochastic reachability techniques to analyze problems in motion planning [Mal+17], fishery management and mathematical finance [SL10], and autonomous surveillance [Kar+11].

1.4 Contributions, Publications, and Organization

1.4.1 Contributions

The main focus of this thesis is on the theory and computation of stochastic reachability and optimal control under uncertainty. Specifically, we utilize convex and stochastic optimization, Fourier transforms, and computational geometry to propose scalable, grid-free, and anytime algorithms as solutions for these problems. The main contributions of this dissertation are summarized below.

1. Introduction of forward stochastic reachability as the characterization of the stochasticity of the state of an uncontrolled stochastic system at a future time of interest (Chapter 3). We propose an experimentally validated algorithm to compute the probability measure and sets using computation geometry and Fourier transforms. We also utilize forward stochastic reachability to compute keep-out sets and their overapproximations (outer-approximations) for stochastic motion planning problems with rigid-body obstacles that have stochastic dynamics using probabilistic occupancy functions.

2. Underapproximation (inner-approximation) of the stochastic reach sets by exploiting convexity and compactness properties (Chapter 4). We first propose sufficient conditions under which the stochastic reach sets, associated with the stochastic reachability problem of a target tube (Figure 1.5), are closed, compact, and convex. Next, we propose a grid-free and scalable approach that uses a ray shooting algorithm to compute polytopic underapproximations of the stochastic reach sets. Our approach can utlize any point-based stochastic reachability algorithm in its computation. We also discuss an interpolative scheme for the stochastic reach sets that remains underapproximative and enables run-time verification.

3. Development of four scalable algorithms to solve the stochastic reachability problem of a target tube for a discrete-time stochastic linear (potentially time-varying) system from a given initial state (Chapter 5). Specifically, for open-loop controller synthesis, we propose a Fourier transform-based approach and discuss significant computational improvements to the existing approaches of particle control and convex chance-constraints. We also utilize these algorithms to construct polytopic underapproximations of the stochastic reach sets. We also present a novel chance-constrained approach using difference of convex optimization for synthesis of affine controllers.

4. Development of `SReachTools`, an open-source MATLAB toolbox to implement all of these algorithms in a user-friendly and repeatability-oriented manner. (Chapter 6)

1.4.2 Publications

All of the work presented in this thesis is published or submitted for publication in peer-reviewed journals and conferences. Publications that are currently under review are marked with 's'. Here, * refers to works with equal contribution.

The work on forward stochastic reachability and probabilistic occupancy functions, which is discussed in Chapter 3, is covered in:

(J1s) A. Vinod and M. Oishi, "Probabilistic occupancy function and sets using forward stochastic reachability for rigid-body dynamic obstacles," submitted to *IEEE Transactions on Automatic Control*, 2018.

(C1) A. Vinod*, S. Rice*, Y. Mao, M. Oishi, B. Acikmese, " Stochastic motion planning using successive convexification and probabilistic occupancy functions," in Proceedings of the *IEEE Conference on Decision and Control (CDC)*, 2018. (accepted for presentation)

(C2) A. Vinod, B. HomChaudhuri, C. Hintz, A. Parikh, S. Buerger, M. Oishi, G. Brunson, S. Ahmad, R. Fierro, "Multiple pursuer-based intercept via forward stochastic reachability," in Proceedings of the *American Control Conference (ACC)*, pp. 1559–1566, 2018.

(C3) B. HomChaudhuri*, A. Vinod*, M. Oishi, " Computation of forward stochastic reach sets: Application to stochastic, dynamic obstacle avoidance," in Proceedings of *American Control Conference (ACC)*, pp. 4404-4411, 2017.

(C4) A. Vinod, B. HomChaudhuri, M. Oishi, "Forward stochastic reachability analysis for uncontrolled linear systems using Fourier transforms," in Proceedings of the *Hybrid Systems: Computation and Control (HSCC)*, pp. 35–44, 2017. (**Best student paper award**)

The work on stochastic reachability of a target tube, which is discussed in Chapters 4 and 5, is covered in:

(J2s) A. Vinod and M. Oishi, "Stochastic reachability of a target tube: Theory and computation," submitted to *IEEE Transactions on Automatic Control*, 2018.

(J3) A. Vinod and M. Oishi, "Scalable underapproximation for stochastic reach-avoid problem for high-dimensional LTI systems using Fourier transforms," In *IEEE Control Systems Letters (L-CSS)*, pp. 316–321, 2017. (Also selected for presentation at the *IEEE Conference on Decision and Control (CDC)*, 2017)

(C5s) A. Vinod and M. Oishi, "Affine controller synthesis for stochastic reachability via difference of convex programming," submitted to Proceedings of the *Hybrid Systems: Computation and Control (HSCC)*, 2019.

(C6s) H. Sartipizadeh, A. Vinod, B. Acikmese, and M. Oishi, "Voronoi partition-based scenario reduction for fast sampling-based stochastic reachability computation of LTI systems," submitted to Proceedings of the *American Control Conference (ACC)*, 2019.

(C7) A. Vinod and M. Oishi, " Scalable underapproximative verification of stochastic LTI systems using convexity and compactness," in Proceedings of the *Hybrid Systems: Computation and Control (HSCC)*, pp. 1–10, 2018.
(**Finalist for the best paper award**)

The toolbox `SReachTools`, that is discussed in Chapter 6, is covered in:

(C8s) A. Vinod, J. Gleason and M. Oishi, "SReachTools: A MATLAB stochastic reachability toolbox," submitted to Proceedings of the *Hybrid Systems: Computation and Control (HSCC)*, 2019.

Although not discussed in this dissertation, we have also developed set-theoretic approaches for stochastic reachability. Referred to as Lagrangian methods, this approach extends existing results for robust stochastic reachability via computational geometry [BR71] to underapproximate stochastic reach sets.

(C9) J. Gleason*, A. Vinod*, and M. Oishi, "Underapproximation of reach-avoid sets for discrete-time stochastic systems via Lagrangian methods," in Proceedings of *IEEE Conference on Decision and Control (CDC)*, pp. 4283-4290, 2017.

(C10) J. Gleason, A. Vinod, and M. Oishi, " Viable set approximation for linear-Gaussian systems with unknown, bounded variance," in Proceedings of *IEEE Conference on Decision and Control (CDC)*, pp. 7049-7055, 2016.

Another work omitted from this dissertation is our work on systems involving human-in-the-loop. Specifically, we have analyzed the user-interface problem, i.e., *what information should be given to a human who works with an automation to collaboratively control a human-automation system to complete a task?* We used discrete optimization techniques, control theory, and human factor guidelines to address this question in a scalable approach. As a numerical example, we constructed user-interfaces for power grid operators using a IEEE-118 bus system. We have also used stochastic

reachability to validate cognitive mental models for humans. These works are covered in:

(C11) A. Vinod, T. Summers, M. Oishi, "User-interface design for MIMO LTI human-automation systems through sensor placement," in Proceedings of *American Control Conference (ACC)*, pp. 5276-5283, 2016.

(C12) A. Vinod, Y. Tang, M. Oishi, K. Sycara, C. Lebiere, and M. Lewis, "Validation of cognitive models for collaborative hybrid systems with discrete human input," in Proceedings of *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pp. 3339–3346, 2016

1.4.3 Organization

In Chapter 2, we present an overview of various mathematical concepts used in this dissertation. We briefly review relevant concepts from real analysis, probability theory, and optimization. We also discuss various system models and controller structures considered in this dissertation.

In Chapter 3, we present a scalable method to perform forward stochastic reachability analysis, that is, a method to compute the forward stochastic reachable set as well as its probability measure. We show that Fourier transforms can be used to provide exact reachability analysis, for systems with bounded or unbounded disturbances and does not require the disturbance to be Gaussian. Further, we provide both iterative and analytical expressions for the probability density, and show that our approach retrieves the Kalman filter's prediction step for the case of Gaussian-perturbed linear systems. We discuss sufficient conditions for the log-concavity of the probability density function, and convexity of the reach set. We also present the theory and algorithms for computing probabilistic occupancy functions and the keep-out sets which are useful in stochastic motion planning problems. Our scalable and grid-free algorithms exploit our proposed sufficient conditions for convexity and compactness

of the keep-out sets. We apply these algorithms to the problem of stochastic target capture and stochastic motion planning. We also present expermimental validation of our stochastic target capture solution on a quadrotor testbed.

In Chapter 4, we consider the problem of stochastic reachability of a target tube. We propose sufficient conditions under which the optimal value functions are Borel measurable, upper semi-continuous, and log-concave, and the stochastic reach set is closed, bounded, compact, and convex. Using these convexity and compactness properties, we describe an underapproximative interpolation technique for the stochastic reach sets. We also consider the problem of synthesis of open-loop and affine feedback controllers to maximize probabilistic safety. Finally, we discuss how these point-based stochastic reachability evaluations can provide underapproximations to the maximal reach probability. These theoretical results enable the construction of tight polytopic underapproximations, that lay the foundations for the design of scalable, grid-free, and anytime algorithms to verify high-dimensional systems.

Chapter 5 builds on the results presented in Chapter 4 to propose four novel approaches for controller synthesis for the stochastic reachability of a target tube. Our approaches are grid-free and scale well with dimension, when compared to the current state-of-the-art dynamic programming-based approaches [Aba+07; Aba+08; SL10]. Our algorithms rely on convex optimization, stochastic programming, and Fourier transforms to compute underapproximations to the maximal reach-probability as well as synthesize open-loop or affine-feedback controllers. We also combine these point-based verification and controller synthesis techniques with a ray-shooting algorithm to underapproximate the sets in a scalable, grid-free, and anytime algorithm. This approach enables, for the first time, the verification of systems as high as 40-dimensions.

Chapter 6 discusses the features of the toolbox, `SReachTools`, that implements all the algorithms presented in this dissertation, and Chapter 7 provides the summary of contributions as well as identifies some of the promising directions for future research.

Chapter 2

Preliminaries

2.1 Notation

We denote the Borel σ-algebra by $\mathscr{B}(\cdot)$, a discrete-time time interval which inclusively enumerates all integers in between a and b for $a, b \in \mathbb{N}$ and $a \leq b$ by $\mathbb{N}_{[a,b]}$, random vectors with bold case, and non-random vectors with an overline. The indicator function of a non-empty set $\mathcal{E}$ is denoted by $1_{\mathcal{E}}(\overline{y})$, such that $1_{\mathcal{E}}(\overline{y}) = 1$ if $\overline{y} \in \mathcal{E}$ and is zero otherwise. We denote $I_n \in \mathbb{R}^n$ as the identity matrix, $\overline{z}_{p\times q}$ as a $p \times q$-dimensional matrix where each element is equal to the scalar $z \in \mathbb{R}$. We define the Minkowski sum as $\oplus$, the Cartesian product of the set $\mathcal{G}$ with itself $k \in \mathbb{N}$ times as $\mathcal{G}^k$, the cardinality of $\mathcal{G}$ with $|\mathcal{G}|$, and the Lebesgue measure of a measurable set $\mathcal{G}$ by $\mathrm{m}(\mathcal{G})$. We define the $\overline{c}$-centered axis-aligned box of side $2a$ as $\mathrm{Box}(\overline{c}, a) = \{\overline{y} \in \mathbb{R}^n : \|\overline{y} - \overline{c}\|_\infty \leq a\}$ where $\overline{c} \in \mathbb{R}^n$ and $a > 0$.

2.2 Real Analysis

2.2.1 Set Properties

This thesis focusses only on Euclidean spaces, denoted by $\mathbb{R}^n$ [Tao06b, Ex. 12.1.6]. We define a Euclidean "open" ball as follows,

$$\text{OpenBall}(\overline{x}, r) = \{\overline{y} \in \mathbb{R}^n : \|\overline{y} - \overline{x}\|_2 < r\} \tag{2.1}$$

Here, $\|\cdot\|_2$ refers to the standard Euclidean norm (l^2 norm) [Tao06b, Ex. 12.1.6]. We define a closed ball as

$$\text{Ball}(\overline{x}, r) = \{\overline{y} \in \mathbb{R}^n : \|\overline{y} - \overline{x}\|_2 \leq r\} \tag{2.2}$$

We provide the definitions of open, closed, bounded, and compact sets [Tao06b, Ch. 12]. A set $\mathcal{E} \subseteq \mathbb{R}^n$ is said to be *open* if and only if for every $\overline{x} \in \mathcal{E}$, there exists $r > 0$ such that $\text{OpenBall}(\overline{x}, r) \subseteq \mathcal{E}$ [Tao06b, Prop. 12.2.15a]. A set $\mathcal{E} \subseteq \mathbb{R}^n$ is *closed* if and only if its complement $\mathbb{R}^n \setminus \mathcal{E} = \{\overline{x} \in \mathbb{R}^n : \overline{x} \notin \mathbb{R}^n\}$ is open [Tao06b, Prop. 12.2.15e]. A set $\mathcal{E} \subseteq \mathbb{R}^n$ is *bounded* if and only if there exists an open ball in $\mathbb{R}^n$ which contains $\mathcal{E}$ [Tao06b, Defn. 12.5.3]. From the Heine-Borel theorem [Tao06b, Thm 12.5.7], $\mathcal{E} \subseteq \mathbb{R}^n$ is *compact* if and only if it is closed and bounded.

Next, we define the interior, boundary, and closure of a set. Let $\mathcal{E} \subseteq \mathbb{R}^n$, and let $\overline{x} \in \mathcal{E}$. The point $\overline{x}$ is an *interior point of* $\mathcal{E}$ if there exists a radius $r > 0$ such that $\text{OpenBall}(\overline{x}, r) \subseteq \mathcal{E}$ [Tao06b, Defn. 12.2.5]. The set of all interior points of $\mathcal{E}$ is called the *interior of* $\mathcal{E}$. The point $\overline{x}$ is a boundary point if and only for every radius $r > 0$, $\text{OpenBall}(\overline{x}, r) \cap \mathcal{E} \neq \emptyset$ and $\text{OpenBall}(\overline{x}, r) \cap (\mathbb{R}^n \setminus \mathcal{E}) \neq \emptyset$ (every open ball has a non-trivial intersection with the set and its complement) [Tao06b, Defn. 12.2.5]. The set of all boundary points of $\mathcal{E}$ is called the *boundary of* $\mathcal{E}$. An open set contains none of its boundary points, while a closed contains all of its boundary points [Tao06b,

Defn. 12.2.12]. The *closure of* $\mathcal{E}$ is the set of all points $\overline{x} \in \mathcal{E}$ such that for every radius $r > 0$, Ball$(\overline{x}, r)$ has a non-empty intersection with $\mathcal{E}$ [Tao06b, Defn. 12.2.9].

Note that these definitions can become trivial in some cases. For example, the set $\{\overline{x} \in \mathbb{R}^3 : x_1 \leq 0, x_2 \leq 0, x_3 = 0\}$ is a hyperplane in $\mathbb{R}^3$. All points on this hyperplane is a boundary point, and it has an empty interior set [BV04, Eg. 2.2]. To avoid this triviality, we define an *affine dimension* as the dimension of the affine hull associated with a set $\mathcal{E} \subseteq \mathbb{R}^n$. Formally, an affine hull of a set $\mathcal{E}$ is the set of all affine combinations of its points,

$$\text{affine}(\mathcal{E}) = \left\{ \sum_{i=1}^{k} \theta_i \overline{x}_i : \overline{x}_i \in \mathcal{E}, \quad \sum_{i=1}^{k} \theta_i = 1 \right\} \tag{2.3}$$

The affine hull affine$(\mathcal{E})$ is the smallest affine set that contains $\mathcal{E}$. The relative interior of a set $\mathcal{E} \subseteq \mathbb{R}^n$ is defined as [BV04, Sec. 2.1.3]

$$\text{relint}(\mathcal{E}) = \{\overline{x} \in \mathbb{R}^n : \exists r > 0, \text{Ball}(\overline{x}, r) \cap \text{affine}(\mathcal{E}) \subseteq \mathcal{E}\}$$

The relative interior of a set is always non-empty, while the interior of a low-dimensional set embedded in a high-dimensional space is empty (no open ball exists such that it is a complete subset of the set). The relative boundary is $\partial\mathcal{E} =$ closure$(\mathcal{E}) \setminus$ relint$(\mathcal{E})$.

A set $\mathcal{E}$ is said to be convex if the line segment joining any two points in $\mathcal{E}$ lies in $\mathcal{E}$ [BV04, Sec. 2.1.4]. Similar to (2.3), we define a convex hull of a set $\mathcal{E}$ as follows [BV04, Sec. 2.1.4],

$$\text{conv}(\mathcal{E}) = \left\{ \sum_{i=1}^{k} \theta_i \overline{x}_i : \overline{x}_i \in \mathcal{E}, \quad \theta_i \geq 0, \quad \sum_{i=1}^{k} \theta_i = 1 \right\}. \tag{2.4}$$

The set conv$(\mathcal{E})$ is the smallest convex set that contains $\mathcal{E}$. Of special interest is the convex hull description of polyhedra. Given a set of points $\overline{x}_i \in \mathbb{R}^n$ with $i \in \mathbb{N}_{[1,M]}$

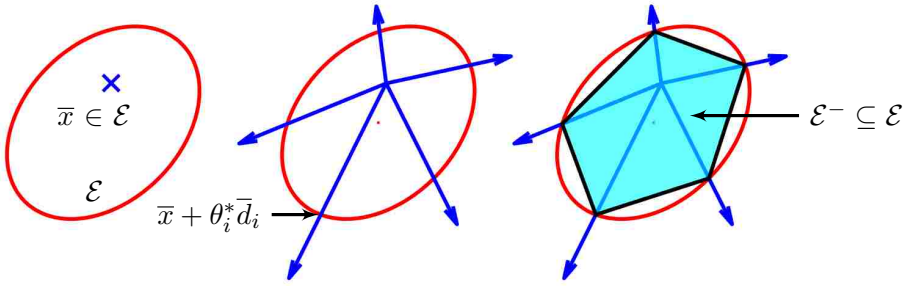


Figure 2.1: Ray-shooting algorithm to underapproximate convex and compact sets.

for some $M \in \mathbb{N}, M > 0$, the convex hull of this finite set is a polytope [BV04, Sec. 2.2.4],

$$\text{conv}_{i \in \mathbb{N}_{[1,M]}}(\overline{x}_i) = \left\{ \sum_{i=1}^{k} \theta_i \overline{x}_i : \theta_i \geq 0, \quad \sum_{i=1}^{k} \theta_i = 1 \right\}. \tag{2.5}$$

A point $\overline{x} \in \mathcal{E}$ is an *extreme point* of the set $\mathcal{E}$ if and only if the only way to express $\overline{x}$ as a convex combination $(1-\theta)\overline{y} + \theta\overline{z}$, such that $\overline{y} \in \mathcal{E}$, $\overline{z} \in \mathcal{E}$, and $0 < \theta < 1$, is by taking $\overline{y} = \overline{z} = \overline{x}$ [Web94, Ch. 2].

Polytopic underapproximation of convex and compact sets has been well studied [BV04, Ex. 2.25] [Web94, Ch. 2]. Figure 2.1 illustrates the *ray-shooting* approach to compute the polytopic underapproximations of convex and compact sets. Given a convex and compact set $\mathcal{E} \subseteq \mathbb{R}^n$, a point $\overline{x} \in \mathcal{E}$, and a set of directions $\overline{d}_i \in \mathbb{R}^n$, $i \in \mathbb{N}_{[1,M]}$, we compute $\theta_i^* > 0$ for each $i \in \mathbb{N}_{[1,M]}$ such that $\overline{x} + \theta_i^* \overline{d}_i \in \partial\mathcal{E}$. The convex hull of these extreme points (2.5) provides a polytope $\mathcal{E}^-$, which is an underapproximation of $\mathcal{E}$ [Web94, Thm 2.6.16]. As expected, the tightness of this underapproximation increases with the number of direction vectors considered. The computation of the boundary points along each of the direction vector requires the computation of θ_i^*. This can be done efficiently via a convex line search (see Sections 2.4.1 and 5.7 and [BV04, Sec. 9.3]).

2.2.2 Function Properties

A function $f : \mathbb{R}^n \to \mathbb{R}$ is upper semi-continuous if its superlevel sets $\{\overline{x} \in \mathbb{R}^n : f(\overline{x}) \geq \alpha\}$ for every $\alpha \in \mathbb{R}$ are closed [Rud87, Defs. 2.3 and 2.8]. Alternatively [BS78, Lem. 7.13b], $f_u(\overline{y})$ is upper semi-continuous if for every sequence $\overline{y}_i \to \overline{y}$, we have

$$\limsup_{i\to\infty} f_u(\overline{y}_i) \leq f_u(\overline{y}). \tag{2.6}$$

A function f is lower semi-continuous if $-f$ is upper semi-continuous, and lower semi-continuous functions have closed sublevel sets $\{\overline{x} \in \mathbb{R}^n : f(\overline{x}) \leq \alpha\}$ for every $\alpha \in \mathbb{R}$. Similar to (2.6), for every upper semi-continuous function $f_u(\cdot)$, $f_l(\overline{y}) \triangleq -f_u(\overline{y})$ is lower semi-continuous, i.e., $\liminf_{i\to\infty}(f_l(\overline{y}_i)) \geq f_l(\overline{y})$ for every sequence $\overline{y}_i \to \overline{y}$ [BS78, Lem. 7.13a]. A function is continuous if and only if it is both upper semi-continuous and lower semi-continuous. The indicator function $1_{\mathcal{E}}(\cdot)$ of a closed set $\mathcal{E}$ is upper semi-continuous.

A function $f : \mathbb{R}^n \to \mathbb{R}$ is convex if it has a convex domain $\mathbf{dom}(f) \subseteq \mathbb{R}^n$ and it satisfies the Jensens' inequality, i.e., for every $\overline{x}, \overline{y} \in \mathbf{dom}(f)$ and $\theta \in [0, 1]$ [BV04, Ch. 3.1.1],

$$f(\theta\overline{x} + (1-\theta)\overline{y}) \leq \theta f(\overline{x}) + (1-\theta) f(\overline{y}). \tag{2.7}$$

A function $f : \mathbb{R}^n \to \mathbb{R}$ is quasiconvex, if its domain and sublevel sets are convex, i.e., $\{\overline{x} \in \mathbf{dom}(f) : f(\overline{x}) \leq \alpha\}$ is convex, for every $\alpha \in \mathbb{R}$. A function f is concave, if $-f$ is convex, and f is quasiconcave, if $-f$ is quasiconvex. A non-negative function $f : \mathbb{R}^n \to \mathbb{R}$ is log-concave if $\log f$ is concave with $\log 0 \triangleq -\infty$ [BV04, Sec. 3.5.1]. Many standard distributions are log-concave, for example, Gaussian, uniform, and exponential [BV04, Eg. 3.40]. The indicator function of a convex set is log-concave (See [BV04, Eg. 3.1 and Sec. 3.1.7]). Since log-concave functions are

quasiconcave [BV04, Sec. 3.5.1], their superlevel sets are convex.

2.3 Probability Theory

A random vector $\boldsymbol{y} \in \mathbb{R}^p$ is a measurable transformation defined in the probability space $(\mathcal{Y}, \mathscr{B}(\mathcal{Y}), \mathbb{P}_{\boldsymbol{y}})$ with sample space $\mathcal{Y} \subseteq \mathbb{R}^p$, (Borel) sigma-algebra $\mathscr{B}(\mathcal{Y})$, and probability measure $\mathbb{P}$ over $\mathscr{B}(\mathcal{Y})$. When $\boldsymbol{y}$ is absolutely continuous, $\boldsymbol{y}$ has a probability density function (PDF) $\psi_{\boldsymbol{y}}$ such that given $\mathcal{G} \in \mathscr{B}(\mathcal{Y})$, $\mathbb{P}_{\boldsymbol{y}}\{\boldsymbol{y} \in \mathcal{G}\} = \int_{\mathcal{G}} \psi_{\boldsymbol{y}}(\overline{y}) d\overline{y}$ where $\overline{y} \in \mathbb{R}^p$, and $\psi_{\boldsymbol{y}}$ is a non-negative Borel measurable function with $\int_{\mathcal{Y}} \psi_{\boldsymbol{y}}(\overline{y}) d\overline{y} = 1$ [CT97, Ch. 1]. For $N \in \mathbb{N}$, a random process is a sequence of random vectors $\{\boldsymbol{y}_k\}_{k=0}^{N}$ where each of the random vectors $\boldsymbol{y}_k$ are defined in the probability space $(\mathbb{R}^p, \mathscr{B}(\mathbb{R}^p), \mathbb{P}_{\boldsymbol{y}_k})$. The random vector $\boldsymbol{Y} = [\boldsymbol{y}_0 \ \boldsymbol{y}_1 \ \dots \ \boldsymbol{y}_N]^\top$ is defined in the probability space $(\mathbb{R}^{p(N+1)}, \mathscr{B}(\mathbb{R}^{p(N+1)}), \mathbb{P}_{\boldsymbol{Y}})$, with $\mathbb{P}_{\boldsymbol{Y}}$ induced from $\mathbb{P}_{\boldsymbol{y}_k}$. See [Gub06; CT97] for details.

The support of $\boldsymbol{y}$, denoted by $\operatorname{supp}(\boldsymbol{y})$, is the smallest closed subset of $\mathcal{Y}$ with probability of occurrence one. Equivalently, from [DJD88, Defn. A.5],

$$\operatorname{supp}(\boldsymbol{y}) = \{\overline{y} \in \mathcal{Y} : \forall r > 0, \ \mathbb{P}_{\boldsymbol{y}}\{\boldsymbol{y} \in \operatorname{Ball}(\overline{y}, r)\} > 0\}. \tag{2.8}$$

From [DJD88, Sec. 2.1], $\mathbb{P}_{\boldsymbol{y}}$ is centrally symmetric if $\mathbb{P}_{\boldsymbol{y}}\{\boldsymbol{y} \in \mathcal{G}\} = \mathbb{P}\{\boldsymbol{y} \in -\mathcal{G}\}$ for every $\mathcal{G} \in \mathscr{B}(\mathcal{Y})$. Recall that semi-continuous functions are Borel-measurable [Tao06b, Lem. 18.5.8].

2.3.1 Log-Concavity in Probability Theory

A probability measure $\mathbb{P}_{\boldsymbol{y}}$ is log-concave if for all convex $\mathcal{G}_A, \mathcal{G}_B \in \mathscr{B}(\mathcal{Y})$ and $\zeta \in [0, 1]$, [Pre80, Sec. 2]

$$\mathbb{P}_{\boldsymbol{y}}\{\boldsymbol{y} \in (\zeta \mathcal{G}_A \oplus (1-\zeta)\mathcal{G}_B)\} \geq \mathbb{P}_{\boldsymbol{y}}\{\boldsymbol{y} \in \mathcal{G}_A\}^{\zeta} \mathbb{P}_{\boldsymbol{y}}\{\boldsymbol{y} \in \mathcal{G}_B\}^{1-\zeta}.$$

Given a convex Borel set $\mathcal{G} \in \mathscr{B}(\mathcal{Y})$ and a log-concave probability measure $\mathbb{P}_{\boldsymbol{y}}$, the following function $h : \mathcal{Y} \to \mathbb{R}$ is log-concave in $\overline{c}$,

$$h(\overline{c}) = \mathbb{P}_{\boldsymbol{y}}\{\boldsymbol{y} \in \{\overline{c}\} \oplus \mathcal{G}\}. \tag{2.9}$$

2.3.2 Fourier Transforms in Probability Theory

For some matrix $H \in \mathbb{R}^{n \times p}$, the stochasticity of the random vector $\boldsymbol{x} = H\boldsymbol{y}$ may be characterized using Fourier transforms. The *characteristic function* of $\boldsymbol{y} \in \mathbb{R}^p$ is

$$\Psi_{\boldsymbol{y}}(\overline{\gamma}) = \mathbb{E}_{\boldsymbol{y}}\left[\exp\left(j\overline{\gamma}^\top \boldsymbol{y}\right)\right] = \int_{\mathbb{R}^p} e^{j\overline{\gamma}^\top \overline{z}} \psi_{\boldsymbol{y}}(\overline{z}) d\overline{z} = \mathscr{F}\left\{\psi_{\boldsymbol{y}}(\cdot)\right\}(-\overline{\gamma}) \tag{2.10}$$

where $\mathscr{F}\{\cdot\}$ denotes the Fourier transformation operator and $\overline{\gamma} \in \mathbb{R}^p$. Since PDFs are absolutely integrable, every PDF has a unique characteristic function [Bil95, Pg. 382]. The characteristic function of the random vector $\boldsymbol{x}$ is then given by

$$\Psi_{\boldsymbol{x}}(\overline{\eta}) = \Psi_{\boldsymbol{y}}(H^\top \overline{\eta}) \tag{2.11}$$

with Fourier variable $\overline{\eta} \in \mathbb{R}^n$ [VHO17, Sec. 2.1]. We can obtain the PDF of $\boldsymbol{x}$ via inverse Fourier transform, provided $\Psi_{\boldsymbol{x}}$ is absolutely integrable [SW71, Cor. 1.21], square integrable [SW71, Thm. 2.4], or Schwartz [SS03, Ch. 6, Thm. 2.4],

$$\psi_{\boldsymbol{x}}(\overline{z}) = \mathscr{F}^{-1}\left\{\Psi_{\boldsymbol{x}}(\cdot)\right\}(-\overline{z}) = \left(\frac{1}{2\pi}\right)^n \int_{\mathbb{R}^n} e^{-j\overline{\eta}^\top \overline{z}} \Psi_{\boldsymbol{x}}(\overline{\eta}) d\overline{\eta}. \tag{2.12}$$

Here, $\mathscr{F}^{-1}\{\cdot\}$ denotes the inverse Fourier transformation operator and $d\overline{\eta}$ is short for $d\eta_1 d\eta_2 \dots d\eta_p$. Recall that Schwartz functions are infinitely differentiable function on $\mathbb{R}^p$ such that the function and its derivatives decrease rapidly [SS03, Sec. 6.2]. Alternatively, we can compute the probability that $\boldsymbol{x}$ lies in a given hypercuboid, $\int_{a_1}^{b_1} \dots \int_{a_n}^{b_n} \psi_{\boldsymbol{x}}(\overline{z}) d\overline{z}$ for any $a_i < b_i \in \mathbb{R}$ with $i \in \mathbb{N}_{[1,n]}$, from its characteristic function

$\Psi_{\boldsymbol{x}}$ by Levy's inversion theorem [CT97, Sec. 8.5, Thm. 1].

2.4 Optimization

2.4.1 Convex Programming

A convex optimization problem has the form:

$$\underset{\overline{z}}{\text{minimize}} \quad f_0(\overline{z}) \tag{2.13a}$$

$$\text{subject to} \quad f_i(\overline{z}) \leq 0 \quad \forall i \in \mathbb{N}_{[1,L]} \tag{2.13b}$$

$$A\overline{z} = b \tag{2.13c}$$

where $\overline{z} \in \mathbb{R}^n$ is the decision variable, the functions $f_i : \mathbb{R}^n \to \mathbb{R}$ are convex for every $i \in \mathbb{N}_{[0,L]}$ with $L \in \mathbb{N}$, $L > 0$, and appropriately dimensioned matrix A and vector b. A convex optimization problem has the useful property that every one of its local minima is also a global minima, which provides significant computational benefits. See [BV04, Ch. 4] for more details.

2.4.2 Difference of Convex Programming

A difference of convex program has the form:

$$\underset{\overline{z}}{\text{minimize}} \quad f_0(\overline{z}) - g_0(\overline{z}) \tag{2.14a}$$

$$\text{subject to} \quad f_i(\overline{z}) - g_i(\overline{z}) \leq 0 \quad \forall i \in \mathbb{N}_{[1,L]} \tag{2.14b}$$

where $\overline{z} \in \mathbb{R}^n$ is the decision variable, the functions $f_i : \mathbb{R}^n \to \mathbb{R}$ and $g_i : \mathbb{R}^n \to \mathbb{R}$ are convex for every $i \in \mathbb{N}_{[0,L]}$, and $L \in \mathbb{N}$, $L > 0$. While (2.14) in general is non-convex, it becomes convex when $g_i(\cdot)$ are affine functions [LB16][HPT00, Chap. 4].

The difference of convex programs form a very broad class of optimization prob-

lems, as any twice differentiable function can be expressed as a difference of convex function [Har59]. For example, $\sin x$ can be equivalently expressed as $(x^2+\sin x)-x^2$; both functions individually are convex since their Hessians are non-negative [BV04, Sec. 3.1.4]. A bilinear constraint of the form $2xy \leq z$ with $x, y, z \in \mathbb{R}$ can be equivalently expressed as a difference of convex constraint $(x+y)^2 - (x^2+y^2+z) \leq 0$ by completing the squares. Here, the functions $(x+y)^2$ and x^2+y^2+z are convex in (x, y, z) [BV04, Sec. 3.1].

Difference of convex programs can be solved to global optimality via general branch-and-bound methods [HPT00]. However, these methods typically require a lot of computational effort. The penalty convex-concave procedure is a successive convexification-based method to find local optima of (2.14) using convex optimization [LB16, Alg. 3.1]. We summarize this procedure in Algorithm 1. It relies on the observation that replacing $g_i(\cdot)$ with their first-order Taylor series approximations in (2.14) yields a convex subproblem, which can then be solved iteratively. To accommodate a potentially infeasible starting point, we relax the difference of convex constraints using slack variables $\overline{\delta}^{(k)} = [\delta_1^{(k)}\ \delta_2^{(k)}\ \ldots\ \delta_L^{(k)}]^\top \in \mathbb{R}^L$, and penalize the value of the slack variables for each iteration k. A possible exit condition, apart from reaching the maximum number of iterations $\tau > \tau_{\max}$, is

$$\left|(f_0(\overline{z}_k) - g_0(\overline{z}_k)) - (f_0(\overline{z}_{k+1}) - g_0(\overline{z}_{k+1})) + \tau_k \sum_{i=1}^{L} (\delta_i^k - \delta_i^{k+1})\right| \leq \epsilon_{\text{dc}} \tag{2.15a}$$

$$\sum\nolimits_{i=1}^{L} \delta_i^{k+1} \leq \epsilon_{\text{viol}} \approx 0 \tag{2.15b}$$

where $\epsilon_{\text{dc}} > 0$ and $\epsilon_{\text{viol}} > 0$ are (small) user-specified tolerances. Here, (2.15a) checks if the algorithm has converged, and (2.15b) checks if $\overline{z}_{k+1}$ is feasible. See [LB16] for more details, such as convergence guarantees of Algorithm 1.

Algorithm 1 Local optimization of (2.14) [LB16, Alg. 3.1]

Input: Initial point $\overline{z}_0$, $\tau_0 > 0$, $\tau_{\max}$, $\gamma > 1$
Output: Local optima of (2.14)

1: $k \leftarrow 0$
2: **do**
3: Convexify: $\hat{g}_i(\overline{z}; \overline{z}_k) \leftarrow g_i(\overline{z}_k) + \nabla g_i(\overline{z}_k)^\top (\overline{z} - \overline{z}_k),\ \forall i \in \mathbb{N}_{[1,L]}$
4: Solve the convex subproblem (2.16) for $\overline{z}_{k+1}, \overline{\delta}^{(k)}$:

$$\begin{aligned} \text{minimize} \quad & f_0(\overline{z}_{k+1}) - \hat{g}_0(\overline{z}_{k+1}; \overline{z}_k) + \tau_k \sum\nolimits_{i=1}^{L} \delta_i^{(k)} \\ \text{subject to} \quad & \overline{\delta}^{(k)} \succeq 0 \\ & f_i(\overline{z}_{k+1}) - \hat{g}_i(\overline{z}_{k+1}; \overline{z}_k) \leq \delta_i^{(k)},\ \forall i \in \mathbb{N}_{[1,L]} \end{aligned} \tag{2.16}$$

5: Update $\tau_{k+1} \leftarrow \min(\gamma \tau_k, \tau_{\max})$ and $k \leftarrow k + 1$
6: **while** $\tau \leq \tau_{\max}$ and (2.15) is not satisfied

2.5 System and Controller Models

2.5.1 Continuous-State System Models

Consider the discrete-time nonlinear time-varying system,

$$\boldsymbol{x}_{k+1} = f_k(\boldsymbol{x}_k, \overline{u}_k, \boldsymbol{w}_k) \tag{2.17}$$

with state $\boldsymbol{x}_k \in \mathcal{X} \subseteq \mathbb{R}^n$, input $\overline{u}_k \in \mathcal{U} \subseteq \mathbb{R}^m$, disturbance $\boldsymbol{w}_k \in \mathcal{W} \subseteq \mathbb{R}^p$, time-varying nonlinear function $f_k : \mathcal{X} \times \mathcal{U} \times \mathcal{W} \to \mathcal{X}$, an initial state $\overline{x}_0 \in \mathcal{X}$, and a time horizon of interest $N \in \mathbb{N}, N > 0$. We assume the input space $\mathcal{U}$ to be compact. We model the disturbance process $\{\boldsymbol{w}_k\}_{k=0}^{N-1}$ in (2.17) as an independent (potentially time-varying) random process. We will assume that the support of $\boldsymbol{w}_k$ is $\mathcal{W}$ for every k. Specifically, we associate with the random vector $\boldsymbol{w}_k$, a probability space $(\mathcal{W}, \mathscr{B}(\mathcal{W}), \mathbb{P}_{\boldsymbol{w},k})$ and a probability density function $\psi_{\boldsymbol{w},k}$ for each $k \in \mathbb{N}_{[0,N-1]}$. The concatenated disturbance random vector $\boldsymbol{W} = [\boldsymbol{w}_0^\top\ \boldsymbol{w}_1^\top\ \ldots\ \boldsymbol{w}_{N-1}^\top]^\top$ is defined in the probability space $(\mathcal{W}^N, \mathscr{B}(\mathcal{W}^N), \mathbb{P}_{\boldsymbol{W}})$ with $\mathbb{P}_{\boldsymbol{W}} = \prod_{k=0}^{N-1} \mathbb{P}_{\boldsymbol{w},k}$. When $\boldsymbol{w}_k$ is not an independent random process, we assume the stochasticity of $\boldsymbol{W}$ is directly available.

We require the nonlinear function f to be Borel-measurable, which guarantees that the state $\{\boldsymbol{x}\}_{k=1}^{N}$ is a random process by (2.17) [CT97, Sec. 1.4, Thm. 4]. Two special cases of (2.17) are

1. affine-perturbed nonlinear time-varying systems,

$$\boldsymbol{x}_{k+1} = g_k(\boldsymbol{x}_k, \overline{u}_k) + \boldsymbol{w}_k \tag{2.18}$$

   where $g_k : \mathcal{X} \times \mathcal{U} \to \mathcal{X}$ is a nonlinear function defined for $k \in \mathbb{N}_{[0,N-1]}$, and

2. linear time-varying systems,

$$\boldsymbol{x}_{k+1} = A_k\boldsymbol{x}_k + B_k\overline{u}_k + \boldsymbol{w}_k \tag{2.19}$$

   where $A_k \in \mathbb{R}^{n\times n}$ and $B_k \in \mathbb{R}^{n\times m}$ are the time-varying state and input matrices defined for $k \in \mathbb{N}_{[0,N-1]}$. For (2.19), the state space is $\mathcal{X} = \mathbb{R}^n$. For linear systems of the form, $\boldsymbol{x}_{k+1} = A_k\boldsymbol{x}_k + B_k\overline{u}_k + F_k\boldsymbol{v}_k$ for some random vector $\boldsymbol{v}_k$ and appropriately dimensioned matrix F_k, one can obtain (2.19) by defining the disturbance random vector $\boldsymbol{w}_k = F_k\boldsymbol{v}_k$

The system (2.17) can be equivalently described by a controlled Markov process with a stochastic kernel that is a time-varying Borel-measurable function $Q_k : \mathscr{B}(\mathcal{X})\times \mathcal{X}\times\mathcal{U} \to [0,1]$. The stochastic kernel assigns a probability measure on the Borel space $(\mathcal{X}, \mathscr{B}(\mathcal{X}))$ for $\boldsymbol{x}_{k+1}$, parameterized by the current state $\boldsymbol{x}_k$ and current action $\overline{u}_k$, i.e., for any $\mathcal{G} \in \mathscr{B}(\mathcal{X})$, $\overline{x} \in \mathcal{X}$, and $\overline{u} \in \mathcal{U}$,

$$\mathbb{P}_{\boldsymbol{x}}\left\{\boldsymbol{x}_{k+1} \in \mathcal{G} | \boldsymbol{x}_k = \overline{x}, \overline{u}_k = \overline{u}\right\} = \mathbb{P}_{\boldsymbol{w},k}\left\{f_k(\overline{x}, \overline{u}, \boldsymbol{w}_k) \in \mathcal{G}\right\} = \int_{\mathcal{G}} Q_k(d\overline{y}|\overline{x}, \overline{u}). \tag{2.20}$$

By (2.20), for any bounded Borel-measurable $h : \mathcal{X} \to \mathbb{R}$,

$$\int_{\mathcal{X}} h(\overline{y}) Q_k(d\overline{y}|\overline{x}, \overline{u}) = \int_{\mathcal{X}} h(f_k(\overline{x}, \overline{u}, \overline{w})) \psi_{\boldsymbol{w},k}(\overline{w}) d\overline{w}. \tag{2.21}$$

In some cases, Q_k may also be explicitly expressed in terms of $\psi_{\boldsymbol{w},k}$,

$$Q_k(d\overline{y}|\overline{x}, \overline{u}) = \begin{cases} \psi_{\boldsymbol{w},k}(\overline{y} - g_k(\overline{x}, \overline{u}))d\overline{y}, & \text{for (2.18)}, \\ \psi_{\boldsymbol{w},k}(\overline{y} - A_k\overline{x} - B_k\overline{u})d\overline{y}, & \text{for (2.19)}. \end{cases} \tag{2.22}$$

We define a *Markov policy* $\pi = (\mu_0, \mu_1, \ldots, \mu_{N-1}) \in \mathcal{M}$ as a sequence of universally measurable state-feedback laws $\mu_k : \mathcal{X} \to \mathcal{U}$ [Aba+08, Defn. 2]. The random vector $\boldsymbol{X} = [\boldsymbol{x}_1^\top \ \boldsymbol{x}_2^\top \ \ldots \ \boldsymbol{x}_N^\top]^\top$, defined in $(\mathcal{X}^N, \mathscr{B}(\mathcal{X}^N), \mathbb{P}_{\boldsymbol{X}}^{\overline{x}_0,\pi})$ has a probability measure $\mathbb{P}_{\boldsymbol{X}}^{\overline{x}_0,\pi}$ defined using Q_k [BS78, Prop. 7.45]. Borel-measurable functions are universally measurable [BS78, Defn. 7.20].

2.5.2 General and Markov policies

We define a *Markov policy* $\pi = (\mu_0, \mu_1, \ldots, \mu_{N-1}) \in \mathcal{M}$ as a sequence of universally measurable functions $\mu_k : \mathcal{X} \to \mathcal{U}$ [Aba+08, Defn. 2][BS78, Defn. 8.4]. A Markov policy is a *current state*-feedback control law since the control at time k is $\overline{u}_k = \mu_k(\overline{x}_k)$.

In stochastic optimal control literature, a more general class of policies is the class of randomized and history-dependent policies [BS78, Defn. 8.4]. We refer to such a policy as a *general policy*. For a general policy π', the control at time k is given by a stochastic kernel defined over $\mathcal{U}$, parameterized by the history of the state and control up to time k, $(\overline{x}_0, \overline{u}_0, \overline{x}_1, \ldots, \overline{u}_{k-1}, \overline{x}_k)$. Markov policies, by definition, are a special class of general policies.

Under the policy π or π', the concatenated state random vector $\boldsymbol{X} = [\boldsymbol{x}_1^\top \ \boldsymbol{x}_2^\top \ \ldots \ \boldsymbol{x}_N^\top]^\top$ has a probability measure $\mathbb{P}_{\boldsymbol{X}}^{\overline{x}_0,\pi}$ or $\mathbb{P}_{\boldsymbol{X}}^{\overline{x}_0,\pi'}$, respectively, induced from $\mathbb{P}_{\boldsymbol{W}}$ and the system dynamics (2.17).

2.5.3 Affine Feedback Controllers

We consider affine feedback controllers for discrete-time linear time-varying systems (2.19). Affine disturbance feedback has been studied in the context of robust and stochastic model predictive control [GKM06; Mes16; FGS16; Old+14; VT11]. An affine disturbance feedback controller is given by

$$\boldsymbol{U} = [\overline{u}_0^\top \ \boldsymbol{u}_1^\top \ \dots \ \boldsymbol{u}_{N-1}^\top]^\top = \overline{M}\boldsymbol{W} + \overline{D} \tag{2.23a}$$

$$\overline{M} = \begin{bmatrix} 0 & \cdots & \cdots & 0 \\ M_{1,0} & 0 & \ddots & 0 \\ \vdots & \ddots & \ddots & \vdots \\ M_{N-1,0} & \cdots & M_{N-1,N-2} & 0 \end{bmatrix}, \overline{D} = \begin{bmatrix} \overline{d}_0 \\ \overline{d}_1 \\ \vdots \\ \overline{d}_{N-1} \end{bmatrix} \tag{2.23b}$$

with the concatenated input vector $\boldsymbol{U} \in \mathbb{R}^{mN}$, matrices $M_{i,j} \in \mathbb{R}^{m \times p}$, the affine disturbance feedback gain matrix $\overline{M} \in \mathbb{R}^{mN \times pN}$, and affine disturbance feedback controller bias $\overline{D} \in \mathbb{R}^{pN}$. The structure of $\overline{M}$ ensures causality. From (2.23), the input $\boldsymbol{u}_k$ at time k depends only on the past disturbances, i.e.,

$$\boldsymbol{u}_k = \sum\nolimits_{j=0}^{k-1} M_{k,j}\boldsymbol{w}_j + \overline{d}_k, \quad \forall k \in \mathbb{N}_{[1,N-1]} \tag{2.24}$$

Thus, an affine disturbance feedback controller is a non-randomized history-dependent policy [BS78, Defn. 8.4], and the input at time $k = 0$, $\overline{u}_0 = \overline{d}_0$, is deterministic. We still denote the control at k via bold-faced symbols to acknowledge that (4.2) is analyzed at $k = 0$ with no information regarding the realization of the disturbances.

Consider the scenario where the concatenated disturbance random vector $\boldsymbol{W} = [\boldsymbol{w}_0^\top \ \boldsymbol{w}_1^\top \ \dots \ \boldsymbol{w}_{N-1}^\top]^\top$ is a Gaussian random vector. Specifically, $\boldsymbol{W} \sim \mathcal{N}(\overline{\mu}_{\boldsymbol{W}}, C_{\boldsymbol{W}})$ with $\overline{\mu}_{\boldsymbol{W}} \in \mathbb{R}^{pN}$ and $C_{\boldsymbol{W}} \in \mathbb{R}^{pN \times pN}$; $C_{\boldsymbol{W}}$ is positive semi-definite, and $\mathbb{P}_{\boldsymbol{W}}$ denotes the probability measure of $\boldsymbol{W}$. Recall that affine transformations of Gaussian random vectors are Gaussian [Gub06, Ch. 9.2]. For a matrix $\Gamma \in \mathbb{R}^{n_y \times (pN)}$ ($n_y \in \mathbb{N}$) and

vector $\overline{\nu} \in \mathbb{R}^{n_y}$,

$$\boldsymbol{W} \sim \mathcal{N}\left(\overline{\mu}_{\boldsymbol{W}}, C_{\boldsymbol{W}}\right) \xrightarrow{\boldsymbol{Y}=\Gamma \boldsymbol{W}+\overline{\nu}} \boldsymbol{Y} \sim \mathcal{N}\left(\overline{\mu}_{\boldsymbol{Y}}, C_{\boldsymbol{Y}}\right) \tag{2.25}$$

with $\overline{\mu}_Y = \Gamma \overline{\mu}_{\boldsymbol{W}} + \overline{\nu} \in \mathbb{R}^{n_y}$ and $C_{\boldsymbol{Y}} = \Gamma C_{\boldsymbol{W}} \Gamma^\top \in \mathbb{R}^{n_y \times n_y}$. Due to the linearity of the system (2.19), we know that

$$\boldsymbol{X} = \bar{A}\overline{x}_0 + H\boldsymbol{U} + E\boldsymbol{W} \tag{2.26}$$

The matrices $\bar{A}$, H, and E are obtained from (2.19) [SB10; VO18a]. By (2.23), (2.25), and (2.26), we know that the concatenated state vector $\boldsymbol{X}$ and $\boldsymbol{U}$ are Gaussian random vectors. Specifically, for a fixed affine disturbance feedback controller $(\overline{M}, \overline{D})$, we have

$$\boldsymbol{X} \sim \mathcal{N}\left(\overline{\mu}_{\boldsymbol{X}}^{\overline{x}_0, \overline{M}, \overline{D}}, C_{\boldsymbol{X}}^{\overline{M}}\right), \tag{2.27a}$$

$$\overline{\mu}_{\boldsymbol{X}}^{\overline{x}_0, \overline{M}, \overline{D}} = \bar{A}\overline{x}_0 + H\overline{D} + (H\overline{M} + E)\overline{\mu}_{\boldsymbol{W}}, \quad C_{\boldsymbol{X}}^{\overline{M}} = (H\overline{M} + E) C_{\boldsymbol{W}} (H\overline{M} + E)^\top \tag{2.27b}$$

and

$$\boldsymbol{U} \sim \mathcal{N}\left(\overline{\mu}_{\boldsymbol{U}}^{\overline{M}, \overline{D}}, C_{\boldsymbol{U}}^{\overline{M}}\right), \tag{2.28a}$$

$$\overline{\mu}_{\boldsymbol{U}}^{\overline{M}, \overline{D}} = \overline{M}\overline{\mu}_{\boldsymbol{W}} + \overline{D}, \quad C_{\boldsymbol{U}}^{\overline{M}} = \overline{M} C_{\boldsymbol{W}} \overline{M}^\top. \tag{2.28b}$$

In other words, the probability measures $\mathbb{P}_{\boldsymbol{X}}^{\overline{x}_0, \overline{M}, \overline{D}}$ and $\mathbb{P}_{\boldsymbol{U}}^{\overline{M}, \overline{D}}$ associated with $\boldsymbol{X}$ and $\boldsymbol{U}$ respectively, are induced from $\mathbb{P}_{\boldsymbol{W}}$ and the affine disturbance feedback controller $(\overline{M}, \overline{D})$.

Similarly, one can consider an affine state feedback controller,

$$\boldsymbol{U} = \overline{K} \begin{bmatrix} \overline{x}_0 \\ \boldsymbol{X} \end{bmatrix} + \overline{V} \tag{2.29}$$

with a block lower-triangular state feedback gain matrix $\overline{K} \in \mathbb{R}^{mN \times (n+1)N}$ and controller bias vector $\overline{V} \in \mathbb{R}^{mN}$. However, affine state feedback and affine disturbance feedback parameterizations of the feedback controllers are equivalent, with straightforward linear transformations between the parameterizations [GKM06, Thm. 9]. Therefore, we will focus only on affine disturbance feedback controllers due to the simplicity in (2.27) and (2.28).

Due to its affine nature, affine disturbance feedback controllers can not satisfy hard control bounds when the disturbance is Gaussian (unbounded). In practice, given a affine disturbance feedback controller, a nonlinear saturation function may be imposed on the affine controller to satisfy the hard control bounds [Hok+12].

Chapter 3

Forward Stochastic Reachability: Theory and Computation

3.1 Introduction

This chapter discusses a new mathematical framework to predict the stochasticity of the state of the system at a future time, given an initial state — the problem of *forward stochastic reachability*. Specifically, we are interested in two properties associated with the state of a given discrete-time stochastic system at a future time of interest:

1. its associated probability measure (the forward stochastic reach probability measure) or density, and

2. its support, the smallest closed set that covers all the reachable states (the forward stochastic reach set).

These properties, illustrated in Figure 1.3, help us answer questions of the form: *what is the likelihood that the system at a future time of interest will lie in a target set? Is there a non-zero likelihood of the state lying in a given collection of states at a future time of interest?*

We discuss a grid-free approach that relies on Fourier transforms and computa-

tional geometry, and scales well with system dimension and time. We are motivated by the problem of autonomous surveillance and stochastic motion planning. We show that the problem of capturing a stochastic target using a controlled robot with deterministic dynamics can be formulated as a convex optimization problem for tractable, globally optimal solutions. For avoiding rigid-body obstacles with stochastic dynamics, we define *probabilistic occupancy functions* which help us reason about collision probability. The superlevel sets of these functions characterize the keep-out sets for probabilistic safety. We discuss sufficient conditions under which these keep-out sets are convex and compact, enabling computationally efficient and grid-free algorithms to characterize these sets using convex optimization. In this thesis, we will restrict our discussion to forward stochastic reachability analysis of linear systems. These results have been extended to Markov jump linear systems in [VO18a].

3.2 Related Work

Note that the forward stochastic reach probability density and reach set have been analyzed separately in control theory literature, for some special cases. For Gaussian-perturbed linear systems, the probability measure at a future time of interest can be obtained from the prediction steps of a Kalman filter [DCA94]. However, this approach fails to generalize for non-Gaussian disturbances, since it tracks only the first two moments of the state. Similarly, for LTI systems with bounded disturbances, established verification methods [Kva+15; KV06; Gir05] can be adapted to overapproximate the forward stochastic reachable set. However, these methods return a trivial result with unbounded disturbances and do not address the forward stochastic reach probability measure, which provides the likelihood of reaching a given set of states. Alternatively, one can use grid-based dynamic programming approaches to compute these quantities which do not scale well with system dimension [Aba+07].

Our work in forward stochastic reachability is motivated by two applications — the problems of capture of a stochastic target and motion planning under stochastic uncertainty (Figure 1.3). The problem of pursuit of a dynamic, non-adversarial target [Hol+09] is relevant in the rescue of a lost first responder in a building on fire [KRS04], capture of a non-aggressive UAV in an urban environment [Gey08], or other non-antagonistic situations. Solutions for an adversarial target, based in a two-person, zero-sum differential game, can accommodate bounded disturbances with unknown stochasticity [MT00; TLS00; Tom+03; BFZ10; Hua+15], but will be conservative for a non-adversarial target. In this chapter, we seek scalable solutions that synthesize an optimal controller for the non-adversarial scenario, by exploiting the forward reachable set and probability measure for the target. We analyze the convexity properties of the forward stochastic reach probability density and sets, and propose a convex optimization problem to provide the exact probabilistic guarantee of success and the corresponding optimal controller.

Stochastic motion planning problems [TBF05; LaV06] require planning a probabilistically safe path for the navigation of a controllable robot in an environment with multiple stochastically moving rigid body obstacles under bounded control authority. Most approaches 1) quantify the collision probability, 2) characterize keep-out regions, the set of states that should be avoided to ensure that the collision probability is below a desired threshold, and 3) generate dynamically-feasible trajectories given a set of keep-out regions, to achieve desired properties like minimizing a performance objective, staying within a safe region, and/or reaching a goal. The first two steps are typically done together using either grid-based approaches [LaV06; TBF05; Elf89; Ich+17; LGSP08; FSL07; BMGF10], chance constraints [BOW11; MW08; Ono+15; LKH10; Aou+13; DTB11], or reachability [ASB09; Sum+11; HVO17; WH12; Mal+17; Chi+15; Chi+17]. The last step may be performed using existing motion planning approaches, such as sampling-based approaches like RRT*

and PRM* [KF11; Mal+17; Chi+15; Chi+17] or optimization-based approaches like mixed-integer linear programming [SFH02; HVO17], mixed-integer quadratic programming [MKK12], and successive convexification [Mao+17]. This chapter provides a grid-free, recursion-free, and sampling-free approach to quantify the collision probability and characterize the keep-out sets through the definition of *probabilistic occupancy function* and *α-probabilistic occupied sets*. The α-probabilistic occupied sets characterizes the keep-out regions to attain a desired probabilistic safety. We propose sufficient conditions for the convexity and compactness of these sets, enabling computationally efficient algorithms.

Grid-based approaches query an occupancy grid [Elf89; TBF05; LaV06] to assess the collision probability. The occupancy grid may be updated using probabilistic velocity obstacles [FSL07], probabilistic inevitable collision state [BMGF10], or by sampling [Ich+17; LGSP08]. Sampling-based approaches (Monte-Carlo simulations) are popular since they can accommodate rigid body obstacles with nonlinear dynamics. This versatility comes at a high computational cost when high-quality approximations are desired [LGSP08; CC06], although importance sampling and the parallelization has improved the computational tractability [Ich+17].

Chance constraints have also been used to plan trajectories for a Gaussian disturbance-perturbed robot navigating an environment with static polytopic obstacles [BOW11; MW08; Ono+15], and extended to obstacles that translate (no rotation) according to a Gaussian process [LKH10; Aou+13]. These approaches replace the probabilistic safety constraints with tighter deterministic constraints that the motion planner must satisfy, and hence are conservative. The probabilistic collision avoidance constraint in [DTB11] for spherical rigid body robot and the obstacles with Gaussian disturbances was formulated as an integral, and an approximation of the keep-out region was provided.

The third approach is to use backward stochastic reachability via dynamic pro-

gramming, to compute the inevitable collision states [Mal+17; Chi+15]. However, these approaches suffer from the curse of dimensionality [Aba+08; Sum+11]. Researchers have proposed particle filters [BOW11; LOE13] and approximate dynamic programming [Kar+14] to improve the computational tractability.

3.3 Problem Statements

This chapter develops the forward stochastic reachability tools to analyze linear dynamics.

Problem 1. *Characterize the forward stochastic reachability for linear dynamics* (2.19)*, i.e., construct analytical expressions for*

1. *the smallest closed set that covers all the reachable states (i.e., the forward stochastic reach set).*

2. *the probability measure over the forward stochastic reach set (i.e., the forward stochastic reach probability measure)*

We will also define a forward stochastic reach probability density, and provide sufficient conditions under which the forward stochastic reach set is convex, and the forward stochastic reach probability density and measure are log-concave. Problem 1 is illustrated in Figure 1.3.

For rigid body obstacles with linear dynamics, we will use forward stochastic reachability to define a probabilistic occupancy function and the α-probabilistic occupied set for a given time of interest. We will seek grid-free, recursion-free, and computationally efficient algorithms to compute the α-probabilistic occupied set by exploiting known results to approximate convex and compact sets.

Problem 2. *Provide algorithms to approximate the α-probabilistic occupied set ($\alpha \in [0, 1]$) for a rigid body obstacle with linear dynamics:*

1. *projection-based tight polytopic approximation, and*

2. *Minkowski sum-based overapproximation.*

Problem 2.a. *Provide sufficient conditions under which the α-probabilistic occupied set of a rigid body obstacle with linear dynamics is convex and compact.*

To address Problem 2.a, we will provide sufficient conditions under which the probabilistic occupancy function is upper semi-continuous and log-concave, and the α-probabilistic occupied set is convex, closed, and bounded.

Finally, we demonstrate the developed theory on two applications — stochastic target capture using forward stochastic reachability (experimentally validated) and stochastic motion planning using α-probabilistic occupied set and successive convexification [Mao+17].

3.4 Forward Stochastic Reachability

Forward stochastic reachability of a system characterizes the stochasticity of the state of a given stochastic system at a future time of interest. It provides the probability measure associated with the state, known as the *forward stochastic reach probability measure*, and the support of the state, known as the *forward stochastic reach set*, at the time of interest.

3.4.1 Forward Stochastic Reachability for Linear Systems

Definitions

Consider the linear system (2.19) initialized to $\overline{x}_0$ and a known open-loop controller $\overline{U}_{N-1} = [\overline{u}_0^\top \ \overline{u}_1^\top \ \dots \ \overline{u}_{N-1}^\top]^\top \in \mathcal{U}^N$ (a zero vector if (2.19) under study is uncontrolled). Recall that the random vector $\boldsymbol{w}_k$ belongs to the probability space $(\mathcal{W}, \mathscr{B}(\mathcal{W}), \mathbb{P}_{\boldsymbol{w},k})$, with $\mathcal{W} \subseteq \mathbb{R}^p$ as the support of $\boldsymbol{w}_k$. Due to the additive disturbance $\boldsymbol{w}_k$, the state

$\boldsymbol{x}_\tau$ at time τ is a random vector. Here, the forward stochastic reach probability measure is the probability measure of $\boldsymbol{x}_\tau$ $\mathbb{P}_{\boldsymbol{x}}^{\tau,\overline{x}_0,\overline{U}_{\tau-1}}$ and the forward stochastic reach set $\mathrm{FSRset}(\tau,\bar{x}_0,\overline{U}_{\tau-1})$ is the support of $\boldsymbol{x}_\tau$ respectively. If a non-negative Borel function $\psi_{\boldsymbol{x}}(\overline{z};\tau,\overline{x}_0,\overline{U}_{\tau-1})$ exists, such that $\int_{\mathcal{X}}\psi_{\boldsymbol{x}}(\overline{z};\tau,\overline{x}_0,\overline{U}_{\tau-1})d\overline{z}=1$, and for any $\mathcal{G}_X\in\mathscr{B}(\mathcal{X})$,

$$\mathbb{P}_{\boldsymbol{x}}^{\tau,\overline{x}_0,\overline{U}_{\tau-1}}\{\boldsymbol{x}_\tau\in\mathcal{G}_X\}=\int_{\mathcal{G}_X}\psi_{\boldsymbol{x}}(\overline{z};\tau,\overline{x}_0,\overline{U}_{\tau-1})dz. \tag{3.1}$$

then $\psi_{\boldsymbol{x}}(\overline{z};\tau,\overline{x}_0,\overline{U}_{\tau-1})$ is the *forward stochastic reach probability density* of the linear system (2.19).

We compactly write the state $\boldsymbol{x}_\tau$ at a time of interest $\tau\geq 1$ as an affine transformation of $\overline{\boldsymbol{w}}_{\tau-1}\triangleq[\boldsymbol{w}_0^\top\ \boldsymbol{w}_1^\top\ \dots\ \boldsymbol{w}_{\tau-1}^\top]^\top$ by separating the elements that evolve under the influence of the stochastic disturbance from those that evolve deterministically. Defining $\overline{x}_{\mathrm{nodist}}(\tau;\overline{U}_{\tau-1},\overline{x}_0)$ as the unperturbed state (disturbance free), we have

$$\boldsymbol{x}_\tau=\underbrace{\mathscr{A}(0,\tau)\overline{x}_0+\mathscr{C}_U(\tau)\overline{U}_{\tau-1}}_{\overline{x}_{\mathrm{nodist}}(\tau;\overline{U}_{\tau-1},\overline{x}_0)}+\mathscr{C}_W(\tau)\overline{\boldsymbol{w}}_{\tau-1} \tag{3.2}$$

Here, the matrices $\mathscr{A}$, $\mathscr{C}_U$, and $\mathscr{C}_W$ are given in (3.3), and are inspired from [SB10, eq. (3)],

$$\mathscr{A}(i,j)=\left(\prod_{k=i}^{j-1}A_k\right)\in\mathbb{R}^{n\times n},\ \text{with}\ \mathscr{A}(i,i)=I_n,\ \text{and}\ i,j\in\mathbb{N},i<j \tag{3.3a}$$

$$\mathscr{C}_U(\tau)=[\mathscr{A}(1,\tau)B_0\ \mathscr{A}(2,\tau)B_2\ \dots\ \mathscr{A}(\tau,\tau)B_{\tau-1}]\in\mathbb{R}^{n\times(m\tau)}, \tag{3.3b}$$

$$\mathscr{C}_W(\tau)=[\mathscr{A}(1,\tau)\ \mathscr{A}(2,\tau)\ \dots\ \mathscr{A}(\tau,\tau)]\in\mathbb{R}^{n\times(p\tau)}. \tag{3.3c}$$

Theorem 1. ***(Forward stochastic reach probability measure characterization)*** *For any time instant $\tau\in\mathbb{N}_{[1,N]}$ and some Borel set $\mathcal{G}_X\in\mathscr{B}(\mathcal{X})$, the forward*

stochastic reach probability measure $\mathbb{P}_{\boldsymbol{x}}^{\tau,\overline{x}_0,\overline{U}_{\tau-1}}$ *of the linear system* (2.19) *is given by*

$$\mathbb{P}_{\boldsymbol{x}}^{\tau,\overline{x}_0,\overline{U}_{\tau-1}}\{\boldsymbol{x}_\tau \in \mathcal{G}_X\} = \mathbb{P}_{\overline{\boldsymbol{w}}}^{\tau-1}\left\{\mathscr{C}_W(\tau)\overline{\boldsymbol{w}}_{\tau-1} \in \left(\mathcal{G}_X \oplus \left\{-\overline{x}_{\text{nodist}}(\tau;\overline{U}_{\tau-1},\overline{x}_0)\right\}\right)\right\}.$$

Proof: Follows from (3.2). ■

Lemma 1. *Given* $\tau \in \mathbb{N}_{[1,N]}$, $\text{supp}(\mathscr{C}_W(\tau)\overline{\boldsymbol{w}}_{\tau-1}) = \mathscr{C}_W(\tau)\mathcal{W}^\tau$.

Proof: Let the singular value decomposition of $\mathscr{C}_W(\tau) \in \mathbb{R}^{n\times(p\tau)}$ be $P\Sigma_C Q^\top$, where $P \in \mathbb{R}^{n\times n}$ and $Q \in \mathbb{R}^{(p\tau)\times(p\tau)}$ are unitary matrices and $\Sigma_C \in \mathbb{R}^{n\times(p\tau)}$ is a diagonal (typically non-square) matrix containing its singular values.

Consider $\text{supp}(P\boldsymbol{a})$ for some n-dimensional random vector $\boldsymbol{a}$ with support $\text{supp}(\boldsymbol{a})$. By (2.8),

$$\begin{aligned}
\text{supp}(P\boldsymbol{a}) &= \{\overline{y} : \forall r > 0, \mathbb{P}_{P\boldsymbol{a}}\{P\boldsymbol{a} \in \text{Ball}(\overline{y}, r)\} > 0\} \\
&= \{\overline{y} : \forall r > 0, \mathbb{P}_{\boldsymbol{a}}\{\boldsymbol{a} \in \text{Ball}(P^{-1}\overline{y}, r)\} > 0\} \qquad (3.4)\\
&= \{P\overline{z} : \forall r > 0, \mathbb{P}_{\boldsymbol{a}}\{\boldsymbol{a} \in \text{Ball}(\overline{z}, r)\} > 0\} = P\text{supp}(\boldsymbol{a}) \qquad (3.5)
\end{aligned}$$

Here, (3.4) follows from [Bil95, Thm. 12.2], while (3.5) follows from the fact that P is a unitary matrix ($|\det(P)| = 1$) and by the change of variables, $P\overline{z} = \overline{y}$. Similarly, $\text{supp}(Q^\top\boldsymbol{b}) = Q^\top\text{supp}(\boldsymbol{b})$ for a $(p\tau)$-dimensional random vector $\boldsymbol{b}$ with support $\text{supp}(\boldsymbol{b})$. Note that

$$\Sigma_C\boldsymbol{b} = (\Sigma_C)_{\text{non-zero,diag}}(\boldsymbol{b})_{\text{non-zero}},$$

where $(\Sigma_C)_{\text{non-zero,diag}}$ is a diagonal square "submatrix" of Σ_C that collects all the non-zero diagonal entries of Σ_C, and $(\boldsymbol{b})_{\text{non-zero}}$ is a random vector constructed from the associated components of the random vector $\boldsymbol{b}$. By arguments similar to (3.5), we have $\text{supp}(\Sigma_C\boldsymbol{b}) = \Sigma_C\text{supp}(\boldsymbol{b})$. Therefore, $\text{supp}(\mathscr{C}_W(\tau)\overline{\boldsymbol{w}}_{\tau-1}) = P\Sigma_C Q^\top\text{supp}(\overline{\boldsymbol{w}}_{\tau-1}) = \mathscr{C}_W(\tau)(\text{supp}(\boldsymbol{w}_k))^\tau$ since $\text{supp}(\overline{\boldsymbol{w}}_{\tau-1}) = (\text{supp}(\boldsymbol{w}_k))^\tau = \mathcal{W}^\tau$. ■

Remark 1. *For a random process* $\{\boldsymbol{w}_k\}$ *with time-varying support* $\mathcal{W}_k$, $\mathrm{supp}(\mathscr{C}_W(\tau)\overline{\boldsymbol{w}}_{\tau-1}) = \mathscr{C}_W(\tau)(\mathcal{W}_0 \times \mathcal{W}_1 \times \ldots \times \mathcal{W}_{\tau-1})$, *by similar arguments as in Lemma 1.*

Theorem 2. ***(Forward stochastic reach set characterization)*** *For* $\tau \in \mathbb{N}_{[1,N]}$,

$$\mathrm{FSRset}(\tau, \bar{x}_0, \overline{U}_{\tau-1}) = \{\overline{x}_{\mathrm{nodist}}(\tau; \overline{U}_{\tau-1}, \overline{x}_0)\} \oplus \mathscr{C}_W(\tau)\mathcal{W}^\tau.$$

Proof: Since $\mathrm{FSRset}(\tau, \bar{x}_0, \overline{U}_{\tau-1}) = \mathrm{supp}(\boldsymbol{x}_\tau)$, we have

$$\begin{aligned}
\mathrm{FSRset}(\tau, \bar{x}_0, \overline{U}_{\tau-1}) &= \left\{\overline{z} \in \mathcal{X} : \forall r > 0,\ \mathbb{P}_{\boldsymbol{x}}^{\tau, \bar{x}_0, \overline{U}_{\tau-1}}\{\boldsymbol{x}_\tau \in \mathrm{Ball}(\overline{z}, r)\} > 0\right\} \\
&= \{\overline{z} \in \mathcal{X} : \forall r > 0, \\
&\qquad \mathbb{P}_{\overline{\boldsymbol{w}}}^{\tau-1}\{\mathscr{C}_W(\tau)\overline{\boldsymbol{w}}_{\tau-1} \in \mathrm{Ball}(\overline{z} - \overline{x}_{\mathrm{nodist}}(\tau; \overline{U}_{\tau-1}, \overline{x}_0), r)\} > 0\big\} \\
&= \big\{\overline{y} = \overline{z} - \overline{x}_{\mathrm{nodist}}(\tau; \overline{U}_{\tau-1}, \overline{x}_0) : \forall r > 0, \\
&\qquad \mathbb{P}_{\overline{\boldsymbol{w}}}^{\tau-1}\{\mathscr{C}_W(\tau)\overline{\boldsymbol{w}}_{\tau-1} \in \mathrm{Ball}(\overline{y}, r)\} > 0\big\}.
\end{aligned}$$

Hence, $\mathrm{FSRset}(\tau, \bar{x}_0, \overline{U}_{\tau-1}) = \mathrm{supp}(\mathscr{C}_W(\tau)\overline{\boldsymbol{w}}_{\tau-1}) \oplus \{\overline{x}_{\mathrm{nodist}}(\tau; \overline{U}_{\tau-1}, \overline{x}_0)\}$. The proof is completed by Lemma 1. ■

Remark 2. *For any* $\mathcal{G} \in \mathscr{B}(\mathcal{X})$, $\mathcal{G} \cap \mathrm{FSRset}(\tau, \bar{x}_0, \overline{U}_{\tau-1}) = \emptyset$ *implies* $\mathbb{P}_{\boldsymbol{x}}^{\tau, \bar{x}_0, \overline{U}_{\tau-1}}\{\boldsymbol{x}_\tau \in \mathcal{G}\} = 0$ *by* (2.8).

Theorems 1 and 2 solve Problem 1. We define the dimension of the FSR set as the dimension of its affine hull [BV04, Sec. 2.1.3]. By Theorem 2, we see that the affine hull of the FSR set is contained in the column space of $\mathscr{C}_W(\tau)$, a subspace of $\mathcal{X}$. Recall that the column space is the range of the linear transformation $\mathscr{C}_W(\tau)$ [Str88, Sec. 2.4]. Clearly, the FSR set is not full-dimensional when column rank of $\mathscr{C}_W(\tau)$ is not n. In such cases, $\boldsymbol{x}_t$ can also be represented as an affine transformation of a lower-dimensional random vector which has a full-dimensional support.

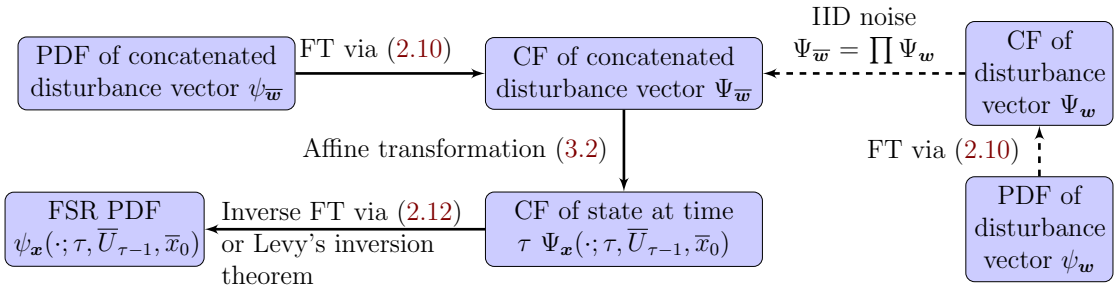


Figure 3.1: Forward stochastic reachability (FSR) using Fourier transforms. Given the probability density function (PDF) of a random vector, we can compute its characteristic function (CF) via Fourier transforms (FT). The dotted lines show how the approach simplifies further, when IID assumption is in place.

3.4.2 Forward Stochastic Reach Probability Density via Fourier Transforms

We have the forward stochastic reach probability density via Theorem 1 and (3.1). We can also use Fourier transforms to compute the probability density function $\psi_{\boldsymbol{x}}(\overline{z};\tau,\overline{x}_0,\overline{U}_{\tau-1})$.

By (2.11), the characteristic function of $\boldsymbol{x}_\tau$, denoted by $\Psi_{\boldsymbol{x}}\left(\overline{\eta};\tau,\overline{U}_{\tau-1},\overline{x}_0\right)$, may be obtained from the characteristic function of $\overline{\boldsymbol{w}}_\tau$, denoted by $\Psi_{\boldsymbol{x}}\left(\overline{\eta};\tau,\overline{U}_{\tau-1},\overline{x}_0\right)$,

$$\Psi_{\boldsymbol{x}}\left(\overline{\eta};\tau,\overline{U}_{\tau-1},\overline{x}_0\right) = e^{j\overline{\eta}^\top \overline{x}_{\mathrm{nodist}}(\tau;\overline{U}_{\tau-1},\overline{x}_0)}\Psi_{\overline{\boldsymbol{w}}}\left(\mathscr{C}_W(\tau)^\top\overline{\eta};\tau\right) \tag{3.6}$$

with Fourier variable $\overline{\eta} \in \mathbb{R}^n$ [VHO17, Sec. 2.1]. The forward stochastic reach probability density can be obtained from (3.6) using Levy's inversion theorem. Alternatively, we can use inverse Fourier transform (2.12), if the density is absolutely integrable, square integrable, or Schwartz. Figure 3.1 shows the Fourier transform approach to compute the forward stochastic reach probability density.

3.4.3 Convexity Properties

Understanding the convexity properties of the forward stochastic reach probability measure and density is useful for tractability. Many standard distributions are log-

concave, including Gaussian, uniform, and exponential distributions [BV04, e.g. 3.39, 3.40].

Lemma 2. ***(Log-concave forward stochastic reach probability measure)*** *If $\mathbb{P}_{\boldsymbol{w}}$ is a log-concave probability measure, then the forward stochastic reach probabilty measure $\mathbb{P}_{\boldsymbol{x}}^{\tau,\bar{x}_0,\overline{U}_{\tau-1}}$ is log-concave over $\mathcal{X}$ for $\forall \tau \in \mathbb{N}_{[1,N]}$.*

Lemma 3. ***(Convex forward stochastic reach set and log-concave probability density)***

a. *Log-concave probabilty measure $\mathbb{P}_{\boldsymbol{x}}^{\tau,\bar{x}_0,\overline{U}_{\tau-1}} \Rightarrow$ convex* $\mathrm{FSRset}(\tau, \bar{x}_0, \overline{U}_{\tau-1})$.

b. *Log-concave probabilty measure $\mathbb{P}_{\boldsymbol{x}}^{\tau,\bar{x}_0,\overline{U}_{\tau-1}}$ and a full-dimensional FSR set* $\mathrm{FSRset}(\tau, \bar{x}_0, \overline{U}_{\tau-1}) \Leftrightarrow$ *log-concave probabilty density $\psi_{\boldsymbol{x}}(\overline{z}; \tau, \bar{x}_0, \overline{U}_{\tau-1})$.*

Lemma 2 follows from Theorem 1 and [DJD88, Lem. 2.1], and Lemma 3 follows from [DJD88, Thms 2.5 and 2.8].

3.5 Applications to Obstacle Avoidance

3.5.1 Properties of a Rigid Body

To simplify analysis, we make the following assumptions (for the discussion in this chapter).

Assumption 1. *The rigid body shape is a Borel set and has a non-zero Lebesgue measure.*

Assumption 2. *The rigid bodies are only allowed to translate.*

Assumption 1 is typically satisfied by real-world problems since open and closed sets are Borel [Rud87, Sec. 1.11], and rigid body obstacle shapes are sets with positive "volume". While Assumption 2 is common practice in motion planning problems [LaV06, Sec. 4.3.2] [Aou+13], it excludes analysis of rigid body obstacles whose

shape has a state-dependent orientation, for example, a unicycle obstacle with a non-centrally symmetric bounded shape that depends on the obstacle's heading. However, by defining an overapproximative shape (that is Borel) that encompasses all attainable shapes, rigid body obstacles that do have state-dependent orientation can be accommodated. For the unicycle obstacle, we can use a large ball that contains the obstacle irrespective of the heading.

We define the set of states "occupied" by the rigid body obstacle given the state of some point in the obstacle (say, center of mass) is $\overline{c} \in \mathcal{X}$ as

$$\mathcal{O}(\overline{c}) = \{\overline{z} \in \mathcal{X} : h_{\text{obs}}(\overline{z} - \overline{c}) \geq 0\} \subseteq \mathcal{X} \tag{3.7}$$

using the zero super-level set of a known Borel-measurable function $h_{\text{obs}} : \mathcal{X} \to \mathbb{R}$. Note that (3.7) is a direct consequence of Assumption 2, since now we can decouple the obstacle geometry from its dynamics. Specifically, the obstacle shape can not be distorted due to its rigidity, and due to absence of rotation, $\mathcal{O}(\overline{c})$ describes the set of occupied states based on the position. For example, we define $h_{\text{obs}}(\overline{z}) = \frac{1}{2} - \|\overline{z}\|_\infty$ for an obstacle whose shape is an axis-aligned hyperbox $\text{Box}(\overline{c}, 1)$ and define $h_{\text{obs}}(\overline{z}) = 1 - \|\overline{z}\|_2$ for an obstacle whose shape is a unit sphere $\text{Ball}(\overline{c}, 1)$.

Lemma 4. *For an obstacle shape $\mathcal{O}(\overline{y})$ with $\overline{y} \in \mathcal{X}$,*

a. (translation invariance) $\mathcal{O}(\overline{y}) = \{\overline{y}\} \oplus \mathcal{O}(\overline{0})$,

b. $-\mathcal{O}(-\overline{y})) = \{\overline{z} \in \mathcal{X} : \overline{y} \in \mathcal{O}(\overline{z})\}$, and

c. $1_{(-\mathcal{O}(-\overline{y}))}(\overline{z}) = 1_{\mathcal{O}(\overline{0})}(\overline{y} - \overline{z})$.

Proof: To show a), define $\overline{z}' = \overline{z} - \overline{y}$. From (3.7), we have $\mathcal{O}(\overline{y}) = \{\overline{z}' + \overline{y} \in \mathcal{X} : h(\overline{z}') \geq 0\} = \{\overline{y}\} \oplus \{\overline{z} : h(z) \geq 0\} = \{\overline{y}\} \oplus \mathcal{O}(\overline{0})$.

Next, we show b) and c). Consider $\overline{y}, \overline{z} \in \mathcal{X}$ such that $\overline{z} \in (-\mathcal{O}(-\overline{y}))$.

$$\begin{aligned}
\overline{z} \in (-\mathcal{O}(-\overline{y})) &\iff -\overline{z} \in \mathcal{O}(-\overline{y}) \\
&\iff h_{\text{obs}}(-\overline{z} - (-\overline{y})) \geq 0 \\
&\iff h_{\text{obs}}(\overline{y} - \overline{z}) \geq 0 \qquad (3.8) \\
&\iff \overline{y} \in \mathcal{O}(\overline{z}) \qquad (3.9) \\
&\iff \overline{y} \in \{\overline{z}\} \oplus \mathcal{O}(\overline{0}) \\
&\iff (\overline{y} - \overline{z}) \in \mathcal{O}(\overline{0}). \qquad (3.10)
\end{aligned}$$

Equation (3.9) shows b), and (3.10) shows c). ■

Lemma 4 provides some useful properties of the set $\mathcal{O}(\overline{c})$ *independent* of the geometric properties of the rigid body like closedness, convexity, and boundedness. Moreover, Lemma 4a shows that it is sufficient to impose geometric restrictions only on $\mathcal{O}(\overline{0})$, due to Assumption 2.

The obstacle shape $\mathcal{O}(\overline{0})$ is centrally symmetric set if $\mathcal{O}(\overline{0}) = -\mathcal{O}(\overline{0})$ [DJD88, Sec. 2.1]. Here, $(-\mathcal{O}(\overline{c}))$ is the reflection of set $\mathcal{O}(\overline{c})$ about origin,

$$-\mathcal{O}(\overline{c}) = \{\overline{z} \in \mathcal{X} : -\overline{z} \in \mathcal{O}(\overline{c})\} = -I_n \mathcal{O}(\overline{c}). \qquad (3.11)$$

We will now apply the developed forward stochastic reachability methods to define a probabilistic occupancy functions and the α-probabilistic occupied set for rigid body obstacles that have linear dynamics. Table 3.1 summarize the results for the probabilistic occupancy function and the α-probabilistic occupied sets discussed in this section.

	Probabilistic occupancy function $\phi_{\boldsymbol{x}}(\overline{y}; \tau, \overline{U}_{\tau-1}, \overline{x}_0, h_{\text{obs}})$	α-probabilistic occupied set $\text{PrOccupySet}(\alpha; \tau, \overline{U}_{\tau-1}, \overline{x}_0, h_{\text{obs}})$
Obstacle dynamics	Linear (2.19)	
Definition	(3.12)–(3.18), Proposition 1	(3.19)
Log-concavity / convexity	Proposition 2	
Upper semi-continuous / closedness	Proposition 4	
Boundedness	-	Proposition 5
Compactness	-	Theorem 3

Table 3.1: Properties of the probabilistic occupancy function and α-probabilistic occupied set established in Section 3.5

3.5.2 Probabilistic Occupancy Function

To define collision, we consider a rigid body robot with shape $\mathcal{R}(\overline{0}) \subseteq \mathcal{X}$ which also satisfies Assumptions 1 and 2. We are motivated by [LaV06, Sec. 4.3.2], where the safety of a trajectory is defined in the *configuration space*. The configuration space encodes the set of transformations that can be applied to a collection of bodies, and it is a state subspace [LaV06, Ch. 7]. Given a collection of obstacles, all configurations in collision with any of the obstacles are removed, and a safe trajectory must exist solely in the remaining space, also known as the *free space*. In the case of stochastic obstacles, the free space becomes a random set [Sum+11], and the safety probability is the probability of a trajectory staying within this random set.

Alternatively, we see that the probability of collision between the rigid body obstacle and the rigid body robot is equivalent to the probability of collision between the robot with exactly same dynamics but shape reduced to a point and the obstacle with an augmented shape $\mathcal{O}(\overline{0}) \oplus (-\mathcal{R}(0))$. This motivates the definition of collision probability, Definition 1, using the probability with which the obstacle "occupies" the state of the point robot. Without loss of generality, we assume the robot shape is a

point and $\mathcal{O}(\overline{0})$ is the appropriately augmented obstacle shape in the sequel.

Definition 1. *A collision is said to occur between a robot and an obstacle, if the robot, described now by a point mass, has the state $\overline{z} \in \mathcal{X}$, the obstacle has state $\overline{c} \in \mathcal{X}$, and $\overline{z} \in \mathcal{O}(\overline{c})$, where $\mathcal{O}(\cdot)$ is now the augmented obstacle shape.*

Consider an obstacle with linear dynamics (2.19) initialized to $\overline{x}_0$ and (if necessary) an known open-loop controller $\overline{U}_{N-1}$. The obstacle shape $\mathcal{O}(\overline{0})$ is defined by $h_{\text{obs}} : \mathcal{X} \to \mathbb{R}$ (see (3.7)). We can define probabilistic occupancy function $\phi_{\boldsymbol{x}}(\overline{y}; \tau, \overline{U}_{\tau-1}, \overline{x}_0, h_{\text{obs}}) : \mathcal{X} \to [0, 1]$ using Lemma 4b as,

$$\phi_{\boldsymbol{x}}(\overline{y}; \tau, \overline{U}_{\tau-1}, \overline{x}_0, h_{\text{obs}}) = \mathbb{P}_{\boldsymbol{x}}^{\tau, \overline{x}_0, \overline{U}_{\tau-1}} \left\{ \boldsymbol{x}_\tau \in \{\overline{z} \in \mathcal{X} : \overline{y} \in \mathcal{O}(\overline{z})\} \right\} \tag{3.12}$$

$$= \mathbb{P}_{\boldsymbol{x}}^{\tau, \overline{x}_0, \overline{U}_{\tau-1}} \left\{ \boldsymbol{x}_\tau \in (-\mathcal{O}(-\overline{y})) \right\}. \tag{3.13}$$

Separately from (3.12) and (3.13), $\phi_{\boldsymbol{x}}$ can also be defined:

1. using Lemma 4a,

$$\phi_{\boldsymbol{x}}(\overline{y}; \tau, \overline{U}_{\tau-1}, \overline{x}_0, h_{\text{obs}}) = \mathbb{P}_{\boldsymbol{x}}^{\tau, \overline{x}_0, \overline{U}_{\tau-1}} \left\{ \boldsymbol{x}_\tau \in -\left(\{-\overline{y}\} \oplus -\mathcal{O}(\overline{0}) \right) \right\}$$

$$= \mathbb{P}_{\boldsymbol{x}}^{\tau, \overline{x}_0, \overline{U}_{\tau-1}} \left\{ \boldsymbol{x}_\tau \in \left(\{\overline{y}\} \oplus -\mathcal{O}(\overline{0}) \right) \right\} \tag{3.14}$$

$$= \mathbb{P}_{\boldsymbol{x}}^{\tau, \overline{x}_0, \overline{U}_{\tau-1}} \left\{ (\overline{y} - \boldsymbol{x}_\tau) \in \mathcal{O}(\overline{0}) \right\} \tag{3.15}$$

2. using expectations ($\mathbb{P}_{\boldsymbol{x}}^{\tau, \overline{x}_0, \overline{U}_{\tau-1}} \{ \boldsymbol{x}_\tau \in \mathcal{G}_X \} = \mathbb{E}_{\boldsymbol{x}}^{\tau, \overline{x}_0, \overline{U}_{\tau-1}} \left[1_{\mathcal{G}_X}(\boldsymbol{x}_\tau) \right]$),

$$\phi_{\boldsymbol{x}}(\overline{y}; \tau, \overline{U}_{\tau-1}, \overline{x}_0, h_{\text{obs}}) = \mathbb{E}_{\boldsymbol{x}}^{\tau, \overline{x}_0, \overline{U}_{\tau-1}} \left[1_{\mathcal{O}(\overline{0})}(\overline{y} - \boldsymbol{x}_\tau) \right] \tag{3.16}$$

3. using convolution (see [HVO17]): By (3.16) and Lemma 4c,

$$\phi_{\boldsymbol{x}}(\overline{y};\tau,\overline{U}_{\tau-1},\overline{x}_0,h_{\text{obs}}) = \int_{\mathcal{X}} \psi_{\boldsymbol{x}}(\overline{z};\tau,\overline{x}_0,\overline{U}_{\tau-1}) 1_{\mathcal{O}(\overline{0})}(\overline{y}-\overline{z}) d\overline{z} \tag{3.17}$$

$$= [\psi_{\boldsymbol{x}} * 1_{\mathcal{O}(\overline{0})}](\overline{y}). \tag{3.18}$$

We obtain a simpler description of $\phi_{\boldsymbol{x}}$ for centrally symmetric obstacles, which follows from (3.11) and (3.15).

Proposition 1. ***($\phi_{\boldsymbol{x}}$ under central symmetry)*** *For a centrally symmetric rigid body obstacle, the probabilistic occupancy function is given by* $\phi_{\boldsymbol{x}}(\overline{y};\tau,\overline{U}_{\tau-1},\overline{x}_0,h_{\text{obs}}) = \mathbb{P}_{\boldsymbol{x}}^{\tau,\overline{x}_0,\overline{U}_{\tau-1}}\{\boldsymbol{x}_\tau \in \mathcal{O}(\overline{y})\}$.

3.5.3 α-Probabilistic Occupied Set

Given $\alpha \in \mathbb{R},\ \alpha \geq 0$, the α-probabilistic occupied set is

$$\text{PrOccupySet}(\alpha;\tau,\overline{U}_{\tau-1},\overline{x}_0,h_{\text{obs}}) = \{\overline{y} \in \mathcal{X} : \phi_{\boldsymbol{x}}(\overline{y};\tau,\overline{U}_{\tau-1},\overline{x}_0,h_{\text{obs}}) \geq \alpha\} \tag{3.19}$$

From (3.19), we have $\forall \tau \in \mathbb{N}_{[1,N]}$ and $\forall \alpha' \in \mathbb{R},\ \alpha \leq \alpha'$,

$$\text{PrOccupySet}(\alpha;\tau,\overline{U}_{\tau-1},\overline{x}_0,h_{\text{obs}}) \supseteq \text{PrOccupySet}(\alpha';\tau,\overline{U}_{\tau-1},\overline{x}_0,h_{\text{obs}}). \tag{3.20a}$$

$$\text{PrOccupySet}(\alpha;\tau,\overline{U}_{\tau-1},\overline{x}_0,h_{\text{obs}}) = \begin{cases} \mathcal{X}, \ \alpha = 0 \\ \emptyset, \ \ \alpha > \phi_{\boldsymbol{x}}(\overline{y}_{\max};\tau,\cdot) \end{cases} \tag{3.20b}$$

where $\overline{y}_{\max} \in \mathcal{X}$ is the maximizer of $\phi_{\boldsymbol{x}}(\overline{y};\tau,\overline{U}_{\tau-1},\overline{x}_0,h_{\text{obs}})$. Equation (3.20b) follows from the fact that $\forall \tau \in \mathbb{N}_{[1,N]}$, $\phi_{\boldsymbol{x}}(\overline{y};\tau,\overline{U}_{\tau-1},\overline{x}_0,h_{\text{obs}})$ is non-negative, and $\phi_{\boldsymbol{x}}(\overline{y}_{\max};\tau,\cdot) \geq \phi_{\boldsymbol{x}}(\overline{y};\tau,\overline{U}_{\tau-1},\overline{x}_0,h_{\text{obs}})$, $\forall \overline{y} \in \mathcal{X}$. By definition, $\phi_{\boldsymbol{x}}(\overline{y}_{\max};\tau,\cdot) \leq 1$.

3.5.4 Properties of Probabilistic Occupancy Function and α-Probabilistic Occupied Set

Proposition 2. ***(Log-concave $\phi_{\boldsymbol{x}}$ and convex*** PrOccupySet***)*** *If $\mathbb{P}_{\boldsymbol{w}}$ is log-concave over $\mathcal{W}$ and $\mathcal{O}(\overline{0})$ is convex, then $\phi_{\boldsymbol{x}}(\overline{y};\tau,\overline{U}_{\tau-1},\overline{x}_0,h_{\text{obs}})$ is log-concave over $\overline{y}$ $\forall\tau\in\mathbb{N}_{[1,N]}$. Moreover,* $\text{PrOccupySet}(\alpha;\tau,\overline{U}_{\tau-1},\overline{x}_0,h_{\text{obs}})$ *is convex $\forall\alpha\in\mathbb{R}$.*

Proof: From Lemma 2, we know the forward stochastic reach probabilty measure $\mathbb{P}_{\boldsymbol{x}}^{\tau,\overline{x}_0,\overline{U}_{\tau-1}}$ is log-concave under these conditions. Note that the set $-\mathcal{O}(\overline{0})$ is convex since $\mathcal{O}(\overline{0})$ is convex and convexity is preserved under linear transformation (see (3.11) and [BV04, Sec. 2.3.2]). Using the definition (3.14) and property (2.9), we conclude that $\phi_{\boldsymbol{x}}(\overline{y};\tau,\overline{U}_{\tau-1},\overline{x}_0,h_{\text{obs}})$ is log-concave over $\overline{y}\in\mathcal{X}$ $\forall\tau\in\mathbb{N}_{[1,N]}$.

The set $\text{PrOccupySet}(\alpha;\tau,\overline{U}_{\tau-1},\overline{x}_0,h_{\text{obs}})$ is convex since log-concave functions are quasiconcave [BV04, Sec. 3.5]. ■

Proposition 2 provides the conditions under which the computation of $\overline{y}_{\max}$ may be posed as an unconstrained log-concave optimization problem, which may be tractably solved when $\phi_{\boldsymbol{x}}$ is given. Alternatively, we can avoid the computation of $\overline{y}_{\max}$ completely by using Proposition 3 for centrally symmetric disturbances, like a zero-mean Gaussian disturbance.

Proposition 3. ***($\overline{y}_{\max}$ under symmetry)*** *If $\mathbb{P}_{\boldsymbol{w}}$ is centrally symmetric and log-concave, and $\mathcal{O}(\overline{0})$ is centrally symmetric and convex, then $\overline{y}_{\max}=\overline{x}_{\text{nodist}}(\tau;\overline{U}_{\tau-1},\overline{x}_0)$.*

Proof: For every $\overline{y}\in\mathcal{X}$, we define $\overline{y}_{\text{shift}}=\overline{y}-\overline{x}_{\text{nodist}}(\tau;\overline{U}_{\tau-1},\overline{x}_0)\in\mathcal{X}$, and $\ell:\mathcal{X}\to[0,1]$,

$$\ell(\overline{y}_{\text{shift}})\triangleq\phi_{\boldsymbol{x}}(\overline{x}_{\text{nodist}}(\tau;\overline{U}_{\tau-1},\overline{x}_0)+\overline{y}_{\text{shift}};\tau,\cdot)=\phi_{\boldsymbol{x}}(\overline{y};\tau,\overline{U}_{\tau-1},\overline{x}_0,h_{\text{obs}}). \tag{3.21}$$

We will show that I) $\ell(\overline{y}_{\text{shift}})$ is even in $\overline{y}_{\text{shift}}\in\mathcal{X}$, II) $\ell(\overline{y}_{\text{shift}})$ is log-concave in $\overline{y}_{\text{shift}}\in\mathcal{X}$, and III) $\overline{0}$ is the maximizer of ℓ since it is log-concave and even. Using

III), $\ell(\overline{0}) \geq \ell(\overline{y}_{\text{shift}}) \equiv \phi_{\boldsymbol{x}}(\overline{x}_{\text{nodist}}(\tau; \overline{U}_{\tau-1}, \overline{x}_0); \tau, \cdot) \geq \phi_{\boldsymbol{x}}(\overline{y}; \tau, \overline{U}_{\tau-1}, \overline{x}_0, h_{\text{obs}})$ for all $\overline{y}_{\text{shift}} \in \mathcal{X}$ and the corresponding $\overline{y}$, which completes the proof.

Part I) ℓ is even over $\overline{y}_{\text{shift}}$: Since $\mathcal{O}(\overline{0})$ is centrally symmetric, we use Proposition 1 to define $\phi_{\boldsymbol{x}}(\overline{y}; \tau, \overline{U}_{\tau-1}, \overline{x}_0, h_{\text{obs}})$. By definition of $\ell(\cdot)$,

$$\begin{aligned}
\ell(\overline{y}_{\text{shift}}) &= \mathbb{P}_{\boldsymbol{x}}^{\tau, \overline{x}_0, \overline{U}_{\tau-1}}\{\boldsymbol{x}_\tau \in \mathcal{O}(\overline{x}_{\text{nodist}}(\tau; \overline{U}_{\tau-1}, \overline{x}_0) + \overline{y}_{\text{shift}})\} \\
&= \mathbb{P}_{\boldsymbol{x}}^{\tau, \overline{x}_0, \overline{U}_{\tau-1}}\{\boldsymbol{x}_\tau \in \mathcal{O}(\overline{0}) \oplus \{\overline{x}_{\text{nodist}}(\tau; \overline{U}_{\tau-1}, \overline{x}_0) + \overline{y}_{\text{shift}}\}\} && (3.22) \\
&= \mathbb{P}_{\overline{\boldsymbol{w}}}^{\tau-1}\{\mathscr{C}_W(\tau)\overline{\boldsymbol{w}}_{\tau-1} \in \mathcal{O}(\overline{0}) \oplus \{\overline{y}_{\text{shift}}\}\} && (3.23) \\
&= \mathbb{P}_{\overline{\boldsymbol{w}}}^{\tau-1}\{\mathscr{C}_W(\tau)\overline{\boldsymbol{w}}_{\tau-1} \in -(\mathcal{O}(\overline{0}) \oplus \{\overline{y}_{\text{shift}}\})\} && (3.24) \\
&= \mathbb{P}_{\overline{\boldsymbol{w}}}^{\tau-1}\{\mathscr{C}_W(\tau)\overline{\boldsymbol{w}}_{\tau-1} \in \mathcal{O}(\overline{0}) \oplus \{-\overline{y}_{\text{shift}}\}\} = \ell(-\overline{y}_{\text{shift}}). && (3.25)
\end{aligned}$$

where (3.22) follows from Lemma 4a, (3.23) follows from (3.2), (3.24) follows from the fact that $\mathbb{P}_{\overline{\boldsymbol{w}}}^{\tau-1}$ is a centrally symmetric probability measure, and (3.25) follows from the fact that $\mathcal{O}(\overline{0})$ is a centrally symmetric set and the definition of $\ell(\overline{y}_{\text{shift}})$ in (3.23).

Part II) ℓ is log-concave over $\overline{y}_{\text{shift}}$: From Proposition 2, we know that $\log\left(\phi_{\boldsymbol{x}}(\overline{y}; \tau, \overline{U}_{\tau-1}, \overline{x}_0, h_{\text{obs}})\right)$ is concave in $\overline{y}$. Since composition of a concave function with an affine function preserves concavity [BV04, Sec. 3.2.2], $\ell(\overline{y}_{\text{shift}})$ is log-concave.

Part III) $\overline{0}$ is the maximizer of ℓ: By definition of log-concavity, $\ell(\overline{0}) \geq \sqrt{\ell(\overline{y}_{\text{shift}})\ell(-\overline{y}_{\text{shift}})}$, $\forall \overline{y}_{\text{shift}} \in \mathcal{X}$ [BV04, Sec. 3.5.1]. Since ℓ is non-negative and $\ell(\overline{y}_{\text{shift}})$ is even in $\overline{y}_{\text{shift}}$, we have $\ell(\overline{0}) \geq \ell(\overline{y}_{\text{shift}})$, $\forall \overline{y}_{\text{shift}} \in \mathcal{X}$. ■

Recall that upper semi-continuous functions are functions with closed superlevel sets (see Section 2.2.2).

Proposition 4. ***(Upper semi-continuous $\phi_{\boldsymbol{x}}$ and closed*** PrOccupySet***)*** *If $\mathcal{O}(\overline{0})$ is closed, then $\forall \tau \in \mathbb{N}_{[1,N]}$, $\phi_{\boldsymbol{x}}(\overline{y}; \tau, \overline{U}_{\tau-1}, \overline{x}_0, h_{\text{obs}})$ is upper semi-continuous, and* $\text{PrOccupySet}(\alpha; \tau, \overline{U}_{\tau-1}, \overline{x}_0, h_{\text{obs}})$ *is closed $\forall \alpha \in \mathbb{R}$.*

Proof: For every sequence $\overline{y}_i \to \overline{y}$, we need to show that

$\limsup_{i\to\infty} \phi_{\boldsymbol{x}}(\overline{y}_i;\tau,\overline{U}_{\tau-1},\overline{x}_0,h_{\text{obs}}) \leq \phi_{\boldsymbol{x}}(\overline{y};\tau,\overline{U}_{\tau-1},\overline{x}_0,h_{\text{obs}})$. By (3.16), this is equivalent to showing $\limsup_{i\to\infty} \mathbb{E}_{\boldsymbol{x}}^{\tau,\overline{x}_0,\overline{U}_{\tau-1}}\left[1_{\mathcal{O}(\overline{0})}(\overline{y}_i-\boldsymbol{x}_\tau)\right] \leq \mathbb{E}_{\boldsymbol{x}}^{\tau,\overline{x}_0,\overline{U}_{\tau-1}}\left[1_{\mathcal{O}(\overline{0})}(\overline{y}-\boldsymbol{x}_\tau)\right]$.

Since $\mathcal{O}(\overline{0})$ is closed, the function $1_{\mathcal{O}(\overline{0})}(\overline{y})$ is upper semi-continuous in $\overline{y}$. The function $-1_{\mathcal{O}(\overline{0})}(\overline{y}-\overline{z})$ is l.s.c in $\overline{y},\overline{z}$ since $-1_{\mathcal{O}(\overline{0})}(\overline{y})$ is l.s.c and $\overline{y}-\overline{z}$ is continuous in $\overline{y},\overline{z}$ [RW09, Ex. 1.4]. Therefore, the function $f:\mathcal{X}^2\to\{0,1\}$ with $f(\overline{y},\overline{z}) = 1-1_{\mathcal{O}(\overline{0})}(\overline{y}-\overline{z})$ is lower semi-continuous in $\overline{y}$ and $\overline{z}$. Specifically, $\liminf_{i\to\infty} f(\overline{y}_i,\overline{z}) \geq f(\overline{y},\overline{z})$ for any $\overline{z}$. Using [Rud87, Thm. 1.12d], we also conclude that f is Borel-measurable in $\overline{y}$ and $\overline{z}$. By construction, $f(\overline{y}_i,\boldsymbol{x}_\tau), f(\overline{y},\boldsymbol{x}_\tau)$ is non-negative (pointwise). Hence, by Fatou's lemma [CT97, Sec. 6.2, Thm. 2.1], the fact that f is l.s.c, Borel-measurable, and non-negative, and linearity of the expectation operator, we have

$$\liminf_{i\to\infty} \mathbb{E}_{\boldsymbol{x}}^{\tau,\overline{x}_0,\overline{U}_{\tau-1}}\left[f(\overline{y}_i,\boldsymbol{x}_\tau)\right] \geq \mathbb{E}_{\boldsymbol{x}}^{\tau,\overline{x}_0,\overline{U}_{\tau-1}}\left[\liminf_{i\to\infty} f(\overline{y}_i,\boldsymbol{x}_\tau)\right] \geq \mathbb{E}_{\boldsymbol{x}}^{\tau,\overline{x}_0,\overline{U}_{\tau-1}}\left[f(\overline{y},\boldsymbol{x}_\tau)\right]$$
$$\therefore \limsup_{i\to\infty} \mathbb{E}_{\boldsymbol{x}}^{\tau,\overline{x}_0,\overline{U}_{\tau-1}}\left[1_{\mathcal{O}(\overline{0})}(\overline{y}_i-\boldsymbol{x}_\tau)\right] \leq \mathbb{E}_{\boldsymbol{x}}^{\tau,\overline{x}_0,\overline{U}_{\tau-1}}\left[1_{\mathcal{O}(\overline{0})}(\overline{y}-\boldsymbol{x}_\tau)\right].$$

Closedness of $\text{PrOccupySet}(\alpha;\tau,\overline{U}_{\tau-1},\overline{x}_0,h_{\text{obs}})$ follows from the upper semi-continuity of $\phi_{\boldsymbol{x}}(\overline{y};\tau,\overline{U}_{\tau-1},\overline{x}_0,h_{\text{obs}})$ and (3.19). ■

Proposition 5. ***(Bounded*** PrOccupySet***)*** *If* $\mathcal{O}(\overline{0})$ *is bounded, then* $\text{PrOccupySet}(\alpha;\tau,\overline{U}_{\tau-1},\overline{x}_0,h_{\text{obs}})$ *is bounded for every* $\alpha>0$ *and* $\tau\in\mathbb{N}_{[1,N]}$.

Proof: Let $b,b_0\in\mathbb{R}$. Consider some $\overline{y}\in\mathcal{X}$ such that $\overline{y}\in \text{PrOccupySet}(\alpha;\tau,\overline{U}_{\tau-1},\overline{x}_0,h_{\text{obs}}) \Rightarrow \phi_{\boldsymbol{x}}(\overline{y};\tau,\overline{U}_{\tau-1},\overline{x}_0,h_{\text{obs}}) > \alpha$. We need to show that for every unit vector $\overline{d}\in\mathcal{X}$, there exists $b_0>0$ such that $\phi_{\boldsymbol{x}}(\overline{y}+b\overline{d};\tau,\cdot)<\alpha$ for all $b>b_0$.

Assume, for contradiction, that there is one unit vector $\overline{d}_c\in\mathcal{X}$ for which no such b_0 exists, or equivalently, for every $b>0$, $\phi_{\boldsymbol{x}}(\overline{y}+b\overline{d}_c;\tau,\cdot)\geq\alpha$.

Since $\mathcal{O}(\overline{0})$ is compact, there exists a ball $\text{Ball}(\overline{0},r)$ centered at origin with radius $r\in\mathbb{R}$, $r>0$ such that $-\mathcal{O}(\overline{0})\subset\text{Ball}(\overline{0},r)$. From Lemma 4a, $-\mathcal{O}(-(\overline{y}+b\overline{d}_c))\subset$

$\text{Ball}(\overline{y} + b\overline{d}_c, r)$ for any $b > 0$. Given $\alpha > 0$, there exists $N_\alpha \in \mathbb{N} \setminus \{0\}$ such that $\alpha N_\alpha > 1$ (Archimedean property [Tao06a, Corr. 5.4.13]). We define $\Gamma_b = \{iR : i \in \mathbb{N}_{[1,N_\alpha]}\} \subset \mathbb{R}$ where $R > 2r$ which is a collection of N_α options for b. By our choice of R, for any $b_1, b_2 \in \Gamma_b,\ b_1 \neq b_2$, the sets $\text{Ball}(\overline{y} + b_1\overline{d}_c; r)$ and $\text{Ball}(\overline{y} + b_2\overline{d}_c; r)$ are distinct, and thereby the sets $-\mathcal{O}(-(\overline{y} + b_1\overline{d}_c))$ and $-\mathcal{O}(-(\overline{y} + b_2\overline{d}_c))$ are distinct. Construct a Borel set $\mathcal{G}_X = \cup_{b\in\Gamma_b}(-\mathcal{O}(-(\overline{y}+b\overline{d}_c)))$, a union of mutually disjoint sets. Hence, $\mathbb{P}_{\boldsymbol{x}}^{\tau,\overline{x}_0,\overline{U}_{\tau-1}}\{\boldsymbol{x}_\tau \in \mathcal{G}_X\} = \sum_{i\in\Gamma_b} \mathbb{P}_{\boldsymbol{x}}^{\tau,\overline{x}_0,\overline{U}_{\tau-1}}\{\boldsymbol{x}_\tau \in (-\mathcal{O}(-(\overline{y}+b\overline{d}_c)))\}$. From (3.13), our assumption on $\overline{d}_c$, and the definition of N_α, we have

$$\mathbb{P}_{\boldsymbol{x}}^{\tau,\overline{x}_0,\overline{U}_{\tau-1}}\{\boldsymbol{x}_\tau \in \mathcal{G}_X\} = \sum_{i\in\Gamma_b} \phi_{\boldsymbol{x}}(\overline{y} + b_i\overline{d}_c; \tau, \cdot) \geq N_\alpha\alpha > 1$$

which leads to a contradiction, completing the proof. ■

We utilize the Heine-Borel theorem and summarize the results from Propositions 2, 4, and 5 as Theorem 3. Theorem 3 addresses Problem 2.a.

Theorem 3. ***(Compact and convex*** PrOccupySet***)*** *If $\mathbb{P}_{\boldsymbol{w}}$ is log-concave over $\mathcal{W}$ and $\mathcal{O}(\overline{0})$ is convex and compact, then* $\text{PrOccupySet}(\alpha; \tau, \overline{U}_{\tau-1}, \overline{x}_0, h_{\text{obs}})$ *is a convex and compact set for all $\alpha > 0$ and $\forall\tau \in \mathbb{N}_{[1,N]}$.*

3.5.5 Polytopic approximation of convex and compact set via projection

We revisit a well-known projection-based approach to compute the polytopic over- and underapproximation of an arbitrary convex and closed set [BV04, Ex. 2.25] [Web94, Ch. 2],

$$\mathcal{L}(\beta) = \{\overline{y} \in \mathbb{R}^n : f(\overline{y}) \geq \beta\} \subseteq \mathbb{R}^n \tag{3.26}$$

for a known upper semi-continuous and quasiconcave function $f : \mathbb{R}^n \to \mathbb{R}$ and known $\beta \in \mathbb{R}$. See Section 2.2.2 for the definitions of upper semi-continuity and

quasiconcavity. Let the maxima of f be larger than β, *i.e.*, $\mathcal{L}(\beta)$ is non-empty. The difference between this approach and the ray shooting algorithm (see Figure 2.1) is the way the boundary points are discovered.

Given $K > 0$ points external to $\mathcal{L}(\beta)$, $\overline{p}_i \in \mathbb{R}^n \setminus \mathcal{L}(\beta),\ i \in \mathbb{N}_{[1,K]}$. We project $\overline{p}_i$ onto $\mathcal{L}(\beta)$ by solving (3.27) for each $i \in \mathbb{N}_{[1,K]}$ [BV04, Sec. 8.1],

$$\begin{array}{ll} \underset{\overline{y}\in\mathbb{R}^n}{\text{minimize}} & \|\overline{y} - \overline{p}_i\|_2 \\ \text{subject to} & f(\overline{y}) \geq \beta \end{array}. \tag{3.27}$$

By construction, (3.27) is a convex optimization problem for each p_i. Further, (3.27) has a unique optimal solution [Web94, Thm. 2.4.1], and we denote this projection point as $P_{\mathcal{L}}(\overline{p}_i) \in \mathcal{L}(\beta)$. We also associate a hyperplane (3.28) with each p_i,

$$\overline{a}_i^\top(\overline{y} - P_{\mathcal{L}}(\overline{p}_i)) \leq 0 \text{ with } \overline{a}_i = \overline{p}_i - P_{\mathcal{L}}(\overline{p}_i). \tag{3.28}$$

Algorithm 2 solves (3.27) for every $\overline{p}_i$ to compute two polytopes $\mathcal{L}_{\text{inner}}(\beta; K)$ and $\mathcal{L}_{\text{outer}}(\beta; K)$,

$$\mathcal{L}_{\text{inner}}(\beta; K) = \text{conv}(P_{\mathcal{L}}(\overline{p}_1), \ldots, P_{\mathcal{L}}(\overline{p}_K)), \tag{3.29}$$

$$\mathcal{L}_{\text{outer}}(\beta; K) = \cap_{i=1}^{K}\{\overline{y} \in \mathbb{R}^n : \overline{a}_i^\top(\overline{y} - P_{\mathcal{L}}(\overline{p}_i)) \leq 0\}. \tag{3.30}$$

See Section 2.2.1 for the definition of convex hulls.

Algorithm 2 Tight polyhedral approximations of $\mathcal{L}(\beta)$ (3.26)

Input: upper semi-continuous and quasiconcave f, $\beta \leq \max_{\overline{y}\in\mathbb{R}^n} f(\overline{y})$, $K > 0$, points $\overline{p}_i \notin \mathcal{L}(\beta)$ for $i \in \mathbb{N}_{[1,K]}$

Output: $\mathcal{L}_{\text{inner}}(\beta; K), \mathcal{L}_{\text{outer}}(\beta; K)$ s.t. $\mathcal{L}_{\text{inner}}(\beta; K) \subseteq \mathcal{L}(\beta) \subseteq \mathcal{L}_{\text{outer}}(\beta; K)$

1: Solve (3.27) for every $i \in \mathbb{N}_{[1,K]}$ to obtain $P_{\mathcal{L}}(\overline{p}_i)$
2: Compute $\mathcal{L}_{\text{inner}}(\beta; K)$ using the convex hull as in (3.29)
3: Compute $\mathcal{L}_{\text{outer}}(\beta; K)$ using the supporting hyperplanes as in (3.30)

Lemma 5. *For a convex, closed, and non-empty set* $\mathcal{L}(\beta)$ *and* K *points* $\overline{p}_i \notin \mathcal{L}(\beta)$, $i \in \mathbb{N}_{[1,K]}$, *Algorithm 2 provides tight polytopic over- and underapproximation, i.e.,* $\mathcal{L}_{\text{inner}}(\beta; K) \subseteq \mathcal{L}(\beta) \subseteq \mathcal{L}_{\text{outer}}(\beta; K)$.

Proof: *Approximation:* The optimization problem (3.27) has a unique optimal solution since $\mathcal{L}(\beta)$ is convex, closed, and non-empty [Web94, Thm. 2.4.1]. The hyperplane in (3.28) is the supporting hyperplane of $\mathcal{L}(\beta)$ at $P_{\mathcal{L}}(\overline{p}_i)$ [BV04, Sec. 2.5.2] [Web94, Thm. 2.4.1]. Note that the set of $P_{\mathcal{L}}(\overline{p}_i)$ is a subset of the extreme points of $\mathcal{L}(\beta)$ and the set of hyperplanes defined using (3.28) is a subset of all the closed halfspaces containing $\mathcal{L}(\beta)$. Hence, we have $\mathcal{L}_{\text{inner}}(\beta; K) \subseteq \mathcal{L}(\beta) \subseteq \mathcal{L}_{\text{outer}}(\beta; K)$ [Web94, Thm 2.6.16, Corr. 2.4.8].

Tightness: Increasing the number of external points $\overline{p}_i$ to $K^+ > K$ (while retaining the previously used external points), we have by the same arguments [Web94, Thm 2.6.16, Corr. 2.4.8] as above,

$$\mathcal{L}_{\text{inner}}(\beta; K) \subseteq \mathcal{L}_{\text{inner}}(\beta; K^+) \subseteq \mathcal{L}(\beta) \subseteq \mathcal{L}_{\text{outer}}(\beta; K^+) \subseteq \mathcal{L}_{\text{outer}}(\beta; K).$$

We thus have a monotone increasing sequence of polytopes in $\mathcal{L}_{\text{inner}}$ and a monotone decreasing sequence of polytopes in $\mathcal{L}_{\text{outer}}$ with increasing K [CT97, Sec. 1]. Therefore,

$$\lim_{K\to\infty} \mathcal{L}_{\text{inner}}(\beta; K) = \cup_{K=1}^{\infty} \mathcal{L}_{\text{inner}}(\beta; K) = \mathcal{L}(\beta), \text{ and}$$
$$\lim_{K\to\infty} \mathcal{L}_{\text{outer}}(\beta; K) = \cap_{K=1}^{\infty} \mathcal{L}_{\text{outer}}(\beta; K) = \mathcal{L}(\beta).$$

■

The computation of $\overline{p}_i$ is easy when $\mathcal{L}(\beta)$ is bounded. For some $\overline{y} \in \mathcal{L}(\beta)$, $\exists r > 0$ such that $\mathcal{L}(\beta) \subseteq \text{Ball}(\overline{y}, r)$. We can now obtain the desired K points that lie outside $\mathcal{L}(\beta)$ by sampling the surface of this ball, denoted by $\partial\text{Ball}(\overline{y}, r)$. For $n \in \{2, 3\}$,

we can uniformly discretize the boundary of a bounding circle/sphere respectively to obtain $\overline{p}_i$. For higher dimensions, we obtain $\overline{p}_i$ by sampling an appropriately dimensioned standard normal distribution, and normalizing the samples to force it to lie on the surface of the bounding hypersphere [HL10]. Computation of the bounding radius r is equivalent to the computation of the diameter of a compact set [Web94, Sec. 2.2], or may be set to a sufficiently large value.

3.5.6 Computation of the α-Probabilistic Occupied Set

We now address Problem 2 and compute approximations of PrOccupySet for a rigid body obstacle with linear dynamics (2.19) initialized to $\overline{x}_0$ and $\overline{U}_{N-1}$. Unfortunately, an exact, closed-form expression for $\phi_{\boldsymbol{x}}$ is typically unavailable. To avoid calculating (3.19) via a grid over the state space $\mathcal{X}$, which is computationally expensive and lacks scalability to higher dimensional systems, we propose two grid-free alternatives to compute PrOccupySet,

1. projection-based tight polytopic approximation (Algorithm 3), and
2. Minkowski sum-based overapproximation (Algorithm 4).

In particular, we seek an overapproximation of $\phi_{\boldsymbol{x}}$ so that we remain conservative with respect to safety, i.e., avoiding the overapproximations of $\phi_{\boldsymbol{x}}$ still provides the desired safety guarantees. An underapproximation of the keep-out regions, when available, provides insight on the degree of conservativeness. These algorithms are recursion-free by the use of forward stochastic reachability via Fourier transforms (see Section 3.4.2).

We can also split the computation of PrOccupySet into an offline and online computation, using Proposition 6 which follows from (3.2), (3.15), and (3.19).

Proposition 6. ***(Effect of*** $\overline{x}_0$ ***on*** $\phi_{\boldsymbol{x}}$ ***and*** PrOccupySet***)*** *Consider the linear system*

(2.19) *initialized to* $\overline{x}_0$ *and* $\overline{U}_{N-1}$. *Then,*

$$\begin{aligned}\phi_{\boldsymbol{x}}(\overline{y};\tau,\overline{U}_{\tau-1},\overline{x}_0,h_{\text{obs}}) &= \phi_{\boldsymbol{x}}\left(\overline{y}-\overline{x}_{\text{nodist}}(\tau);\tau,\overline{U}_{\tau-1},\overline{0}_{n\times 1},h_{\text{obs}}\right)\\ &\quad \text{PrOccupySet}(\alpha;\tau,\overline{U}_{\tau-1},\overline{x}_0,h_{\text{obs}})\\ &\quad = \{\overline{x}_{\text{nodist}}(\tau;\overline{U}_{\tau-1},\overline{x}_0)\}\oplus \text{PrOccupySet}(\alpha;\tau,\overline{U}_{\tau-1},\overline{0}_{n\times 1},h_{\text{obs}})\end{aligned}$$

Proposition 6 allows the computation of $\text{PrOccupySet}(\alpha;\tau,\overline{U}_{\tau-1},\overline{0}_{n\times 1},h_{\text{obs}})$ be done offline, and an online computation of its Minkowski sum with $\overline{x}_{\text{nodist}}(\tau;\overline{U}_{\tau-1},\overline{x}_0)$, which is the translation of $\text{PrOccupySet}(\alpha;\tau,\overline{U}_{\tau-1},\overline{0}_{n\times 1},h_{\text{obs}})$ by $\overline{x}_{\text{nodist}}(\tau;\overline{U}_{\tau-1},\overline{x}_0)$ to obtain $\text{PrOccupySet}(\alpha;\tau,\overline{U}_{\tau-1},\overline{x}_0,h_{\text{obs}})$. Proposition 6 can enable faster planning, especially when solving motion planning problems in an environment with homogeneous rigid body obstacles with stochastic linear dynamics.

Projection-Based Tight Polytopic Approximation

From Theorem 3, we know the sufficient conditions under which PrOccupySet is convex and compact. To compute tight polytopic approximations of PrOccupySet, we replace the generic projection problem (3.27) with

$$\begin{array}{ll}\underset{\overline{y}\in\mathcal{X}}{\text{minimize}} & \|\overline{y}-\overline{p}_i\|_2\\ \text{subject to} & \log(\phi_{\boldsymbol{x}}(\overline{y};\tau,\overline{U}_{\tau-1},\overline{x}_0,h_{\text{obs}})) \;\geq\; \log(\alpha)\end{array} \tag{3.31}$$

Problem (3.31) is also convex, as guaranteed by Proposition 2. Using Algorithm 2, we obtain two polytopes $\text{UnPrOccupySet}(\alpha;\tau,\overline{U}_{\tau-1},\overline{x}_0,h_{\text{obs}})$ and $\text{OvPrOccupySet}(\alpha;\tau,\overline{U}_{\tau-1},\overline{x}_0,h_{\text{obs}})$ such that

$$\begin{aligned}&\text{UnPrOccupySet}(\alpha;\tau,\overline{U}_{\tau-1},\overline{x}_0,h_{\text{obs}})\subseteq \text{PrOccupySet}(\alpha;\tau,\overline{U}_{\tau-1},\overline{x}_0,h_{\text{obs}}),\ \text{and}\\ &\text{OvPrOccupySet}(\alpha;\tau,\overline{U}_{\tau-1},\overline{x}_0,h_{\text{obs}})\supseteq \text{PrOccupySet}(\alpha;\tau,\overline{U}_{\tau-1},\overline{x}_0,h_{\text{obs}}).\end{aligned}$$

In addition, we use (3.20b) to simplify the computation. We summarize this approach in Algorithm 3.

Algorithm 3 Projection-based tight polytopic approximations of $\text{PrOccupySet}(\alpha; \tau, \overline{U}_{\tau-1}, \overline{x}_0, h_{\text{obs}})$

Input: DPV with arbitrary $\boldsymbol{w}$, threshold $\alpha \geq 0$, convex and compact rigid body $\mathcal{O}(\overline{0})$, a desired number of samples K for Algorithm 2, bounding ball radius $r > 0$
Output: $\text{UnPrOccupySet}(\alpha; \tau, \overline{U}_{\tau-1}, \overline{x}_0, h_{\text{obs}}), \text{OvPrOccupySet}(\alpha; \tau, \overline{U}_{\tau-1}, \overline{x}_0, h_{\text{obs}})$

$\overline{y}_{\max} \leftarrow \max_{\overline{y} \in \mathcal{X}} \phi_{\boldsymbol{x}}(\overline{y}; \tau, \overline{U}_{\tau-1}, \overline{x}_0, h_{\text{obs}})$ ▷ Use Prop. 3 when valid
if $\phi_{\boldsymbol{x}}(\overline{y}_{\max}; \tau, \cdot) \leq \alpha$ **then**
    $\text{UnPrOccupySet}(\alpha; \tau, \overline{U}_{\tau-1}, \overline{x}_0, h_{\text{obs}}), \text{OvPrOccupySet}(\alpha; \tau, \overline{U}_{\tau-1}, \overline{x}_0, h_{\text{obs}}) \leftarrow \emptyset$ by (3.20b)
else if $\alpha > 0$ **then**
    Sample K points $\overline{p}_i \in \partial\text{Ball}(\overline{y}_{\max}, r)$
    $\text{UnPrOccupySet}(\alpha; \tau, \overline{U}_{\tau-1}, \overline{x}_0, h_{\text{obs}}), \text{OvPrOccupySet}(\alpha; \tau, \overline{U}_{\tau-1}, \overline{x}_0, h_{\text{obs}}) \leftarrow$ Algorithm 2 with (3.31)
else
    $\text{UnPrOccupySet}(\alpha; \tau, \overline{U}_{\tau-1}, \overline{x}_0, h_{\text{obs}}), \text{OvPrOccupySet}(\alpha; \tau, \overline{U}_{\tau-1}, \overline{x}_0, h_{\text{obs}}) \leftarrow \mathcal{X}$ by (3.20b)
end if

Minkowski Sum-Based Overapproximation

Consider the following set defined using $\mathcal{O}(\overline{0})$ and a superlevel set of $\psi_{\boldsymbol{x}}(\overline{z}; \tau, \overline{x}_0, \overline{U}_{\tau-1})$,

$$\begin{aligned}&\text{PrOccupySet}^{+}(\alpha; \tau, \overline{U}_{\tau-1}, \overline{x}_0, h_{\text{obs}}) \\ &\quad = \left\{ \overline{z} \in \mathcal{X} : \psi_{\boldsymbol{x}}(\overline{z}; \tau, \overline{x}_0, \overline{U}_{\tau-1}) \geq \frac{\alpha}{\text{m}(\mathcal{O}(\overline{0}))} \right\} \oplus \mathcal{O}(\overline{0}). \end{aligned} \tag{3.32}$$

where $\text{m}(\mathcal{O}(\overline{0}))$ refers to the Lebesgue measure of the set $\mathcal{O}(\overline{0})$. Proposition 7 ensures that $\text{PrOccupySet}^{+}(\alpha; \tau, \overline{U}_{\tau-1}, \overline{x}_0, h_{\text{obs}})$ is an overapproximation of the α-probabilistic occupied set.

Proposition 7. *For any $\alpha \in \mathbb{R}, \alpha \geq 0$, $\tau \in \mathbb{N}_{[1,N]}$ and a bounded forward stochastic reach probabilty density $\psi_{\boldsymbol{x}}(\overline{z}; \tau, \overline{x}_0, \overline{U}_{\tau-1})$,* $\text{PrOccupySet}(\alpha; \tau, \overline{U}_{\tau-1}, \overline{x}_0, h_{\text{obs}}) \subseteq \text{PrOccupySet}^{+}(\alpha; \tau, \overline{U}_{\tau-1}, \overline{x}_0, h_{\text{obs}})$.

Proof: Let $\overline{y} \in \text{PrOccupySet}(\alpha; \tau, \overline{U}_{\tau-1}, \overline{x}_0, h_{\text{obs}}) \Rightarrow \phi_{\boldsymbol{x}}(\overline{y}; \tau, \overline{U}_{\tau-1}, \overline{x}_0, h_{\text{obs}}) \geq \alpha$. Recall that for any two non-negative measurable functions f_1, f_2 such that $f_1(\overline{z}) \leq f_2(\overline{z})$ for all $\overline{z} \in \mathcal{X}$, we have $\int_{\mathcal{X}} f_1(\overline{z})d\overline{z} \leq \int_{\mathcal{X}} f_2(\overline{z})d\overline{z}$ [Tao06b, Prop. 19.2.6(c)]. By (3.13),

$$\begin{aligned}\alpha \leq \phi_{\boldsymbol{x}}(\overline{y}; \tau, \overline{U}_{\tau-1}, \overline{x}_0, h_{\text{obs}}) &= \int_{\mathcal{X}} \psi_{\boldsymbol{x}}(\overline{z}; \tau, \overline{x}_0, \overline{U}_{\tau-1}) 1_{(-\mathcal{O}(-\overline{y}))}(\overline{z}) d\overline{z} \\ &\leq \left(\sup_{\overline{z} \in (-\mathcal{O}(-\overline{y}))} \psi_{\boldsymbol{x}}(\overline{z}; \tau, \overline{x}_0, \overline{U}_{\tau-1}) \right) \text{m}\left(-\mathcal{O}(-\overline{y})\right). \quad (3.33)\end{aligned}$$

If the forward stochastic reach probabilty density $\psi_{\boldsymbol{x}}(\overline{z}; \tau, \overline{x}_0, \overline{U}_{\tau-1})$ is unbounded, the upper bound given in (3.33) could be trivially ∞. From Lemma 4a and [Bil95, Thms. 12.1 and 12.2], $\text{m}\left(-\mathcal{O}(-\overline{y})\right) = \text{m}\left(\{\overline{y}\} \oplus (-\mathcal{O}(\overline{0}))\right) = \text{m}\left(\mathcal{O}(\overline{0})\right)$. By Lemma 4b,

$$\alpha \leq \phi_{\boldsymbol{x}}(\overline{y}; \tau, \overline{U}_{\tau-1}, \overline{x}_0, h_{\text{obs}}) \Rightarrow \left(\sup_{\overline{y} \in \mathcal{O}(\overline{z})} \psi_{\boldsymbol{x}}(\overline{z}; \tau, \overline{x}_0, \overline{U}_{\tau-1}) \right) \text{m}\left(\mathcal{O}(\overline{0})\right) \geq \alpha.$$

By Assumption 1, $\text{m}(\mathcal{O}(\overline{0})) \neq 0$. Hence, $\exists \overline{z} \in \mathcal{X}$ such that $\overline{y} \in \mathcal{O}(\overline{z})$ and $\psi_{\boldsymbol{x}}(\overline{z}; \tau, \overline{x}_0, \overline{U}_{\tau-1}) \geq \frac{\alpha}{\text{m}(\mathcal{O}(\overline{0}))}$. From Lemma 4a, $\overline{y} \in (\mathcal{O}(\overline{0}) \oplus \{\overline{z}\})$ where $\overline{z}$ satisfies the condition $\psi_{\boldsymbol{x}}(\overline{z}; \tau, \overline{x}_0, \overline{U}_{\tau-1}) \geq \frac{\alpha}{\text{m}(\mathcal{O}(\overline{0}))}$, which completes the proof. ■

Equation (3.33) is the only overapproximation step in the proof of Proposition 7. This inequality becomes tighter when the "variation" of $\psi_{\boldsymbol{x}}(\overline{z}; \tau, \overline{x}_0, \overline{U}_{\tau-1})$ in $-\mathcal{O}(-\overline{y})$ becomes smaller. For example, when the set $\mathcal{O}(\overline{0})$ is contained in a relatively small ball.

Recall that support functions may be used to characterize a convex and compact set $\mathcal{G} \subset \mathcal{X}$ [Web94, Thm. 5.6.4]. We denote the *support function* of $\mathcal{G}$ by $\rho : \mathcal{X} \to \mathcal{X}$

$$\rho(\overline{l}; \mathcal{G}) = \sup_{\overline{y} \in \mathcal{G}} \overline{l}^{\top} \overline{y}, \qquad \overline{l} \in \mathbb{R}^n. \quad (3.34)$$

For a Gaussian $\boldsymbol{x}_\tau \sim \mathcal{N}(\overline{\mu}_{\boldsymbol{x}_\tau}, \Sigma_{\boldsymbol{x}_\tau})$ and some $\kappa \in \mathbb{R}, \kappa > 0$, the κ-superlevel set of

$\psi_{\boldsymbol{x}}(\overline{z};\tau,\overline{x}_0,\overline{U}_{\tau-1})$ is an ellipsoid $\mathcal{E}(\overline{\mu}_{\boldsymbol{x}_\tau},Q_{\boldsymbol{x}_\tau}(\kappa))$ [KV97, Defn. 2.1.4],

$$\mathcal{E}(\overline{\mu}_{\boldsymbol{x}_\tau},Q_{\boldsymbol{x}_\tau}(\kappa)) \triangleq \left\{\overline{z}\in\mathcal{X} : (\overline{z}-\overline{\mu}_{\boldsymbol{x}_\tau})^\top (Q_{\boldsymbol{x}_\tau}(\kappa))^{-1}(\overline{z}-\overline{\mu}_{\boldsymbol{x}_\tau}) \leq 1\right\} \tag{3.35}$$

with shape matrix $Q_{\boldsymbol{x}_\tau}(\kappa) = -2\log\left(\kappa\sqrt{|2\pi\Sigma_{\boldsymbol{x}_\tau}|}\right)\Sigma_{\boldsymbol{x}_\tau}$. From [KV97, Defn 2.1.4], we have a closed form expression for the support function of ellipsoid (3.35) with some positive definite $Q_{\boldsymbol{x}_\tau}(\kappa)\in\mathbb{R}^{n\times n}$,

$$\rho\left(\overline{l};\mathcal{E}(\overline{\mu}_{\boldsymbol{x}_\tau},Q_{\boldsymbol{x}_\tau}(\kappa))\right) = \overline{l}^\top\overline{\mu}_{\boldsymbol{x}_\tau} + \sqrt{\overline{l}^\top Q_{\boldsymbol{x}_\tau}(\kappa)\overline{l}}. \tag{3.36}$$

Thus, for a Gaussian $\boldsymbol{x}_\tau$, (3.32) simplifies to

$$\text{PrOccupySet}^+(\alpha;\tau,\overline{U}_{\tau-1},\overline{x}_0,h_{\text{obs}}) = \mathcal{E}\left(\overline{\mu}_{\boldsymbol{x}_\tau},Q_{\boldsymbol{x}_\tau}\left(\frac{\alpha}{\text{m}(\mathcal{O}(\overline{0}))}\right)\right)\oplus\mathcal{O}(\overline{0}). \tag{3.37}$$

The support function of the Minkowski sum of two non-empty, convex, and compact sets is the sum of the respective support functions [Web94, Thm. 5.6.2]. Therefore, we have a closed-form description of the support function of $\text{PrOccupySet}^+(\alpha;\tau,\overline{U}_{\tau-1},\overline{x}_0,h_{\text{obs}})$,

$$\begin{aligned}\rho(\overline{l};\text{PrOccupySet}^+(\alpha;\tau,\overline{U}_{\tau-1},\overline{x}_0,h_{\text{obs}})) = \overline{l}^\top\overline{\mu}_{\boldsymbol{x}_\tau} + \sqrt{\overline{l}^\top Q_{\boldsymbol{x}_\tau}\left(\frac{\alpha}{\text{m}(\mathcal{O}(\overline{0}))}\right)\overline{l}}\\ + \rho(\overline{l};\mathcal{O}(\overline{0})).\end{aligned} \tag{3.38}$$

For a polytopic $\mathcal{O}(\overline{0}) = \{\overline{y}\in\mathbb{R}^n : H\overline{y}\leq k\}$ with appropriate H,k, $\rho(\overline{l};\mathcal{O}(\overline{0}))$ is solved via a linear program given $\overline{l}$ [LGG09, Sec. 4.1]. For an ellipsoidal $\mathcal{O}(\overline{0}) = \mathcal{E}(\overline{0},Q_{\mathcal{O}})$ with an appropriate $Q_{\mathcal{O}}\in\mathbb{R}^{n\times n}$, $\rho(\overline{l};\mathcal{O}(\overline{0}))$ is given by (3.36). Alternatively, we can use the ellipsoidal toolbox (`ET`) [GK] to compute ellipsoidal overapproximations of $\text{PrOccupySet}^+(\alpha;\tau,\overline{U}_{\tau-1},\overline{x}_0,h_{\text{obs}})$, as Minkowski sums of ellipsoids need not be ellipsoids [KV97, Pg. 97]. Note that $\text{PrOccupySet}^+(\alpha;\tau,\overline{U}_{\tau-1},\overline{x}_0,h_{\text{obs}})$ is an extension

of the results in [DTB11] for arbitrary robot and obstacle rigid body shapes and Gaussian-perturbed obstacle dynamics.

In general, given a support function, a tight polytopic overapproximation of $\text{PrOccupySet}^+(\alpha;\tau,\overline{U}_{\tau-1},\overline{x}_0,h_{\text{obs}})$ can be found by "sampling" the direction vectors $\bar{l}$ [LGG09, Prop. 3]. Specifically, using the support function of $\text{PrOccupySet}^+(\alpha;\tau,\overline{U}_{\tau-1},\overline{x}_0,h_{\text{obs}})$ given in (3.38), we define a tight polytopic overapproximation

$$\text{OvPrOccupySet}^+(\alpha;\tau,\overline{U}_{\tau-1},\overline{x}_0,h_{\text{obs}}) = \{\overline{y}\in\mathbb{R}^n : A_{\text{des}}\overline{y}\leq\overline{b}\} \tag{3.39}$$

with $N_{\text{des}} > 0$, $A_{\text{des}} = [\overline{a}_1^\top\ \overline{a}_2^\top\ \dots\ \overline{a}_{N_{\text{des}}}^\top]^\top$ with $\overline{a}_i\in\mathbb{R}^n$ as the desired supporting hyperplane directions, and $\overline{b} = [b_1\ b_2\ \dots\ b_{N_{\text{des}}}]^\top\in\mathbb{R}^{N_{\text{des}}}$ with

$$b_i = \rho(\overline{a}_i;\text{PrOccupySet}^+(\alpha;\tau,\overline{U}_{\tau-1},\overline{x}_0,h_{\text{obs}})),\quad \forall i\in\mathbb{N}_{[1,N_{\text{des}}]}. \tag{3.40}$$

Here, tightness refers to the fact that the supporting hyperplanes of the polytope $\text{OvPrOccupySet}^+(\alpha;\tau,\overline{U}_{\tau-1},\overline{x}_0,h_{\text{obs}})$, $\overline{a}_i^\top\overline{y}\leq b_i$, support the set $\text{PrOccupySet}^+(\alpha;\tau,\overline{U}_{\tau-1},\overline{x}_0,h_{\text{obs}})$. Algorithm 4 computes $\text{OvPrOccupySet}^+(\alpha;\tau,\overline{U}_{\tau-1},\overline{x}_0,h_{\text{obs}})$ for a Gaussian $\boldsymbol{x}_\tau$ using A_{des} and (3.40). Algorithm 4 does not require numerical quadrature like Algorithm 3, and works well even when $\mathcal{O}(\overline{0})$ is contained in a relatively small ball.

For a non-Gaussian disturbance $\boldsymbol{w}_k$, the forward stochastic reach probabilty density may be obtained through Fourier transforms (see Section 3.4.2). We can then utilize Algorithm 2 to compute a tight polytopic overapproximation of the level set of the forward stochastic reach probabilty density and, if required, $\mathcal{O}(\overline{0})$. In this case, the set $\text{PrOccupySet}^+(\alpha;\tau,\overline{U}_{\tau-1},\overline{x}_0,h_{\text{obs}})$ is the Minkowski sum of two polytopes, which can be easily computed using the Multi-Parametric Toolbox (`MPT3`) [Her+13].

Algorithm 4 Minkowski sum-based approximation of PrOccupySet for a Gaussian forward stochastic reach probabilty density

Input: DPV with Gaussian disturbance $\boldsymbol{w}$, threshold $\alpha \geq 0$, support function of the rigid body $\rho(\bar{l}; \mathcal{O}(\overline{0}))$, matrix of desired supporting hyperplane directions A_{des}
Output: $\text{OvPrOccupySet}^{+}(\alpha; \tau, \overline{U}_{\tau-1}, \overline{x}_0, h_{\text{obs}})$

$\overline{y}_{\max} \leftarrow \max_{\overline{y} \in \mathcal{X}} \phi_{\boldsymbol{x}}(\overline{y}; \tau, \overline{U}_{\tau-1}, \overline{x}_0, h_{\text{obs}})$ ▷ Use Prop. 3 when valid
if $\phi_{\boldsymbol{x}}(\overline{y}_{\max}; \tau, \cdot) \leq \alpha$ **then**
    $\text{OvPrOccupySet}^{+}(\alpha; \tau, \overline{U}_{\tau-1}, \overline{x}_0, h_{\text{obs}}) \leftarrow \emptyset$ by (3.20b)
else if $\alpha > 0$ **then**
    for $i \in \mathbb{N}_{[1, N_{\text{des}}]}$ **do**
        $b_i \leftarrow \rho(\overline{a}_i; \text{PrOccupySet}^{+}(\alpha; \tau, \overline{U}_{\tau-1}, \overline{x}_0, h_{\text{obs}}))$ by (3.38) and (3.40)
    end for
    $\text{OvPrOccupySet}^{+}(\alpha; \tau, \overline{U}_{\tau-1}, \overline{x}_0, h_{\text{obs}}) \leftarrow \{\overline{y} \in \mathbb{R}^n : A_{\text{des}} \overline{y} \leq \overline{b}\}$
else
    $\text{OvPrOccupySet}^{+}(\alpha; \tau, \overline{U}_{\tau-1}, \overline{x}_0, h_{\text{obs}}) \leftarrow \mathcal{X}$ by (3.20b)
end if

Remark 3. *Proposition 6 holds true for the approximations* $\text{UnPrOccupySet}(\alpha; \tau, \overline{U}_{\tau-1}, \overline{x}_0, h_{\text{obs}})$, $\text{OvPrOccupySet}(\alpha; \tau, \overline{U}_{\tau-1}, \overline{x}_0, h_{\text{obs}})$, $\text{PrOccupySet}^{+}(\alpha; \tau, \overline{U}_{\tau-1}, \overline{x}_0, h_{\text{obs}})$, *and* $\text{OvPrOccupySet}^{+}(\alpha; \tau, \overline{U}_{\tau-1}, \overline{x}_0, h_{\text{obs}})$.

Algorithms 3 and 4 can provide higher quality approximations at the cost of computational time. For Algorithm 3, using more external points $\overline{p}_i$ in Algorithm 2 (increasing K) based on $\text{UnPrOccupySet}(\alpha; \tau, \overline{U}_{\tau-1}, \overline{x}_0, h_{\text{obs}})$ can yield tighter overapproximations of $\text{PrOccupySet}(\alpha; \tau, \overline{U}_{\tau-1}, \overline{x}_0, h_{\text{obs}})$. For Algorithm 4, using a larger set of direction vectors (more rows in A_{des}) tightens the overapproximation of $\text{PrOccupySet}^{+}(\alpha; \tau, \overline{U}_{\tau-1}, \overline{x}_0, h_{\text{obs}})$.

3.6 Application: Stochastic Target Capture with Experimental Validation

3.6.1 Problem Setup

We consider the problem of a controlled robot (R) having to capture a stochastically moving non-adversarial target, denoted here by a goal robot (G). The robot R has controllable linear dynamics while the robot G has uncontrollable linear dynamics, perturbed by an absolutely continuous random vector. The robot R is said to capture robot G if the robot G is inside a pre-determined set defined around the current position of robot R. We seek an *open-loop* controller (independent of the current state of robot G) for the robot R which maximizes the probability of capturing robot G within the time horizon T. The information available to solve this problem are the position of the robots R and G at $t = 0$, the deterministic dynamics of the robot R, the perturbed dynamics of the robot G, and the density of the perturbation. We consider a 2-D environment, but our approach can be easily extended to higher dimensions. We perform the FSR analysis in the inertial coordinate frame.

We model the robot R as a point mass system discretized in time,

$$\bar{x}_R[t+1] = \bar{x}_R[t] + B_R \bar{u}_R[t] \tag{3.41}$$

with state (position) $\bar{x}_R[t] \in \mathbb{R}^2$, input $\bar{u}_R[t] \in \mathcal{U} \subseteq \mathbb{R}^2$, input matrix $B_R = T_s I_2$ and sampling time T_s. From (3.41),

$$\bar{x}_R[\tau+1] = \bar{x}_R[0] + (\bar{1}_{1\times\tau} \otimes B_R)\overline{U}_\tau,\ \tau \in [0, T-1] \tag{3.42}$$

with the open-loop input vector $\overline{U}_\tau = [\bar{u}_R^\top[\tau-1]\ \bar{u}_R^\top[\tau-2]\ \ldots\ \bar{u}_R^\top[0]]^\top$, $\overline{U}_\tau \in \mathcal{U}^\tau \subseteq \mathbb{R}^{(2\tau)}$.

We consider two cases for the dynamics of the robot G: 1) point mass dynamics,

and 2) double integrator dynamics, both discretized in time and perturbed by an absolutely continuous random vector. In the former case, we presume that the velocity is drawn from a bivariate Gaussian distribution,

$$\boldsymbol{x}_G[t+1] = \boldsymbol{x}_G[t] + B_{\mathrm{G,PM}}\boldsymbol{v}_G[t] \tag{3.43a}$$

$$\boldsymbol{v}_G[t] \sim \mathcal{N}(\bar{\mu}_G^{\boldsymbol{v}}, \Sigma_G). \tag{3.43b}$$

The state (position) is the random vector $\boldsymbol{x}_G[t]$ in the probability space $(\mathcal{X}, \sigma(\mathcal{X}), \mathbb{P}_{\boldsymbol{x}_G}^{t,\bar{x}_G[0]})$ with $\mathcal{X} = \mathbb{R}^2$, disturbance matrix $B_{\mathrm{G,PM}} = B_R$, and $\bar{x}_G[0]$ as the known initial state of the robot G. The stochastic velocity $\boldsymbol{v}_G[t] \in \mathbb{R}^2$ has mean vector $\bar{\mu}_G^{\boldsymbol{v}}$ and covariance matrix Σ_G. In the latter case, acceleration in each direction is an independent exponential random variable,

$$\boldsymbol{x}_G[t+1] = A_{\mathrm{G,DI}}\boldsymbol{x}_G[t] + B_{\mathrm{G,DI}}\boldsymbol{a}[t] \tag{3.44a}$$

$$(\boldsymbol{a}[t])_{\mathrm{x}} \sim \mathrm{Exp}(\lambda_{\mathrm{ax}}), \quad (\boldsymbol{a}[t])_{\mathrm{y}} \sim \mathrm{Exp}(\lambda_{\mathrm{ay}}) \tag{3.44b}$$

$$A_{\mathrm{G,DI}} = I_2 \otimes \begin{bmatrix} 1 & T_s \\ 0 & 1 \end{bmatrix}, B_{\mathrm{G,DI}} = I_2 \otimes \begin{bmatrix} \frac{T_s^2}{2} \\ T_s \end{bmatrix}.$$

The state (position and velocity) is the random vector $\boldsymbol{x}_G[t]$ in the probability space $(\mathcal{X}_{\mathrm{DI}}, \sigma(\mathcal{X}_{\mathrm{DI}}), \mathbb{P}_{\boldsymbol{x}_G}^{t,\bar{x}_G[0]})$ with $\mathcal{X}_{\mathrm{DI}} = \mathbb{R}^4$ and $\bar{x}_G[0]$ as the known initial state of the robot G. The stochastic acceleration $\boldsymbol{a}[t] = [(\boldsymbol{a}[t])_{\mathrm{x}}\ (\boldsymbol{a}[t])_{\mathrm{y}}]^\top \in \mathbb{R}^2_+ = [0,\infty) \times [0,\infty)$ has the following probability density and characteristic function ($\bar{z} = [z_1\ z_2]^\top \in \mathbb{R}^2_+ = [0,\infty) \times [0,\infty), \bar{\alpha} = [\alpha_1\ \alpha_2]^\top \in \mathbb{R}^2$),

$$\psi_{\boldsymbol{a}}(\bar{z}) = \lambda_{\mathrm{ax}}\lambda_{\mathrm{ay}} \exp\left(-\lambda_{\mathrm{ax}} z_1 - \lambda_{\mathrm{ay}} z_2\right) \tag{3.45}$$

$$\Psi_{\boldsymbol{a}}(\bar{\alpha}) = \frac{\lambda_{\mathrm{ax}}\lambda_{\mathrm{ay}}}{(\lambda_{\mathrm{ax}} - j\alpha_1)(\lambda_{\mathrm{ay}} - j\alpha_2)}. \tag{3.46}$$

The characteristic function $\Psi_{\boldsymbol{a}}(\bar{\alpha})$ is defined using Property P3 and the characteristic

function of the exponential given in [Bil95, Section 26].

Formally, the robot R captures robot G at time τ if $\boldsymbol{x}_G[\tau] \in \text{CaptureSet}(\bar{x}_R[\tau])$. In other words, the capture region of the robot R is the $\text{CaptureSet}(\bar{y}) \subseteq \mathbb{R}^2$ when robot R is at $\bar{y} \in \mathbb{R}^2$. In other words, the capture probability is

$$\text{CapturePr}_{\bar{x}_R}(\tau, \bar{x}_R[\tau]; \bar{x}_G[0]) = \mathbb{P}^{\tau, \bar{x}_G[0]}_{\boldsymbol{x}_G}\{\boldsymbol{x}_G[\tau] \in \text{CaptureSet}(\bar{x}_R[\tau])\} \tag{3.47}$$

$$= \int_{\text{CaptureSet}(\bar{x}_R[\tau])} \psi_{\boldsymbol{x}_G}(\bar{y}; \tau, \bar{x}_G[0]) d\bar{y}. \tag{3.48}$$

We maximize the capture probability by considering the following optimization problem,

$$\begin{aligned} &\text{maximize} && \text{CapturePr}_{\bar{x}_R}(\tau, \bar{x}_R[\tau]; \bar{x}_G[0]) \\ &\text{subject to} && \begin{cases} \tau & \in [1, T] \\ \bar{x}_R[\tau] & \in \text{Reach}_R(\tau; \bar{x}_R[0]) \end{cases} \end{aligned} \tag{3.49}$$

where the decision variables are the time of capture τ and the position of the robot R $\bar{x}_R[\tau]$ at time τ. From (3.42), we define the reach set for the robot R at time τ as

$$\text{Reach}_R(\tau; \bar{x}_R[0]) = \left\{\bar{y} \in \mathcal{X} | \exists \overline{U}_\tau \in \mathcal{U}^\tau \text{ s.t. } \bar{x}_R[\tau] = \bar{y}\right\} = A_R \bar{x}_R[0] + (\bar{1}_{1\times\tau} \otimes B_R)\mathcal{U}^\tau. \tag{3.50}$$

Note that the original problem of designing an open-loop input vector $\overline{U}_{(\cdot)}$ is implicitly enforced via the reach constraint (3.50). Several deterministic reachability computation tools are available for the computation of $\text{Reach}_R(\tau; \bar{x}_R[0])$, like MPT [Her+13] and ET [KV06].

By (2.9), Lemma 2, and (3.47), we know that $\text{CapturePr}_{\bar{x}_R}(\tau, \bar{x}_R[\tau]; \bar{x}_G[0])$ is log-concave in $\bar{x}_R[\tau]$ for every $\tau \in \mathbb{N}_{[1,N]}$. Further, $\text{Reach}_R(\tau; \bar{x}_R[0])$ is a convex set by

(3.50). Therefore, for any $\tau \in [1, T]$,

$$\begin{aligned} \text{minimize} \quad & -\log(\text{CapturePr}_{\bar{x}_R}(\tau, \bar{x}_R[\tau]; \bar{x}_G[0])) \\ \text{subject to} \quad & \bar{x}_R[\tau] \in \text{Reach}_R(\tau; \bar{x}_R[0]) \end{aligned} \tag{3.51}$$

is convex with the decision variable $\bar{x}_R[\tau]$. Problem 3.51 is an equivalent convex optimization problem of the partial maximization with respect to $\bar{x}_R[\tau]$ of Problem 3.49 since we have transformed the original objective function with a monotone function to yield a convex objective and the constraint sets are identical [BV04, Section 4.1.3].

We solve Problem 3.49 by solving Problem 3.51 for each time instant $\tau \in [1, T]$ to obtain $\bar{x}_R^*[\tau]$ and compute the maximum of the resulting finite set to get $(\tau^*, \bar{x}_R^*[\tau^*])$. Since Problem 3.49 could be non-convex, this approach ensures a global optimum is found.

3.6.2 Goal Robot with Point Mass Dynamics

We solve Problem 3.49 for the system given by (3.43). Here, the disturbance set is $\mathcal{W} = \mathbb{R}^2$. The probability of successful capture of the robot G can be computed using (3.48) since the forward stochastic reach probability density $\psi_{\boldsymbol{x}_G}(\cdot; \tau, \bar{x}_G[0])$ is available.

We implement the problem with the following parameters: $T_s = 0.2$, $T = 20$, $\bar{\mu}_G^{\boldsymbol{v}} = [1.3 \;\; 0.3]^\top$, $\Sigma_G = \begin{bmatrix} 0.5 & 0.8 \\ 0.8 & 2 \end{bmatrix}$, $\bar{x}_G[0] = [-3 \;\; 0]^\top$, $\bar{x}_R[0] = [-3 \;\; -2]^\top$ and $\mathcal{U} = [1, 2]^2$. The capture region of the robot R is a box centered about the position of the robot $\bar{y}$ with edge length $2a$ ($a = 0.25$) and edges parallel to the axes — $\text{CaptureSet}(\bar{y}) = \text{Box}(\bar{y}, a)$, a convex set.

Figure 3.2 (left) shows the evolution of the mean position of the robot G and the optimal capture position for the robot R at time instants $4, 5, 8, 14$, and 20. The contour plots of $\psi_{\boldsymbol{x}_G}(\cdot; \tau, \bar{x}_G[0])$ are rotated ellipses since Σ_E is not a diagonal matrix.

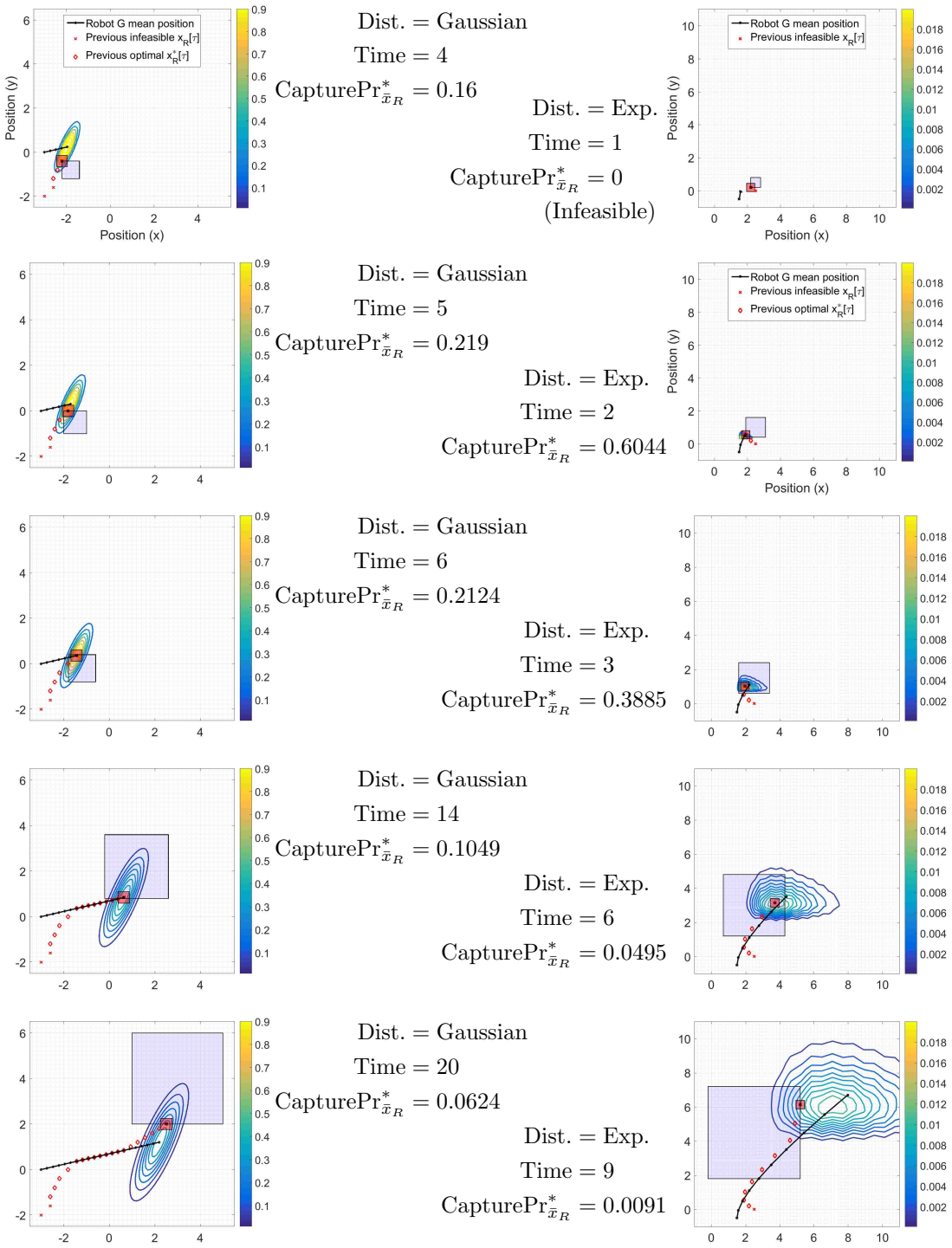


Figure 3.2: Snapshots of optimal capture positions of the robots G and R when G has point mass dynamics (3.43) with Gaussian model for the velocity (left) and double integrator dynamics (3.44) with exponential model for the acceleration (right). The blue line shows the mean position trajectory of robot G $\mu_G[\tau]$, the contour plot characterizes $\psi_{\boldsymbol{x}_G}(\cdot;\tau,\bar{x}_G[0])$, the blue box shows the reach set of the robot R at time τ $\text{Reach}_R(\tau,\bar{x}_R[0])$, and the red box shows the capture region centered at $\bar{x}_R^*[\tau]$ $\text{CaptureSet}(\bar{x}_R^*[\tau])$.

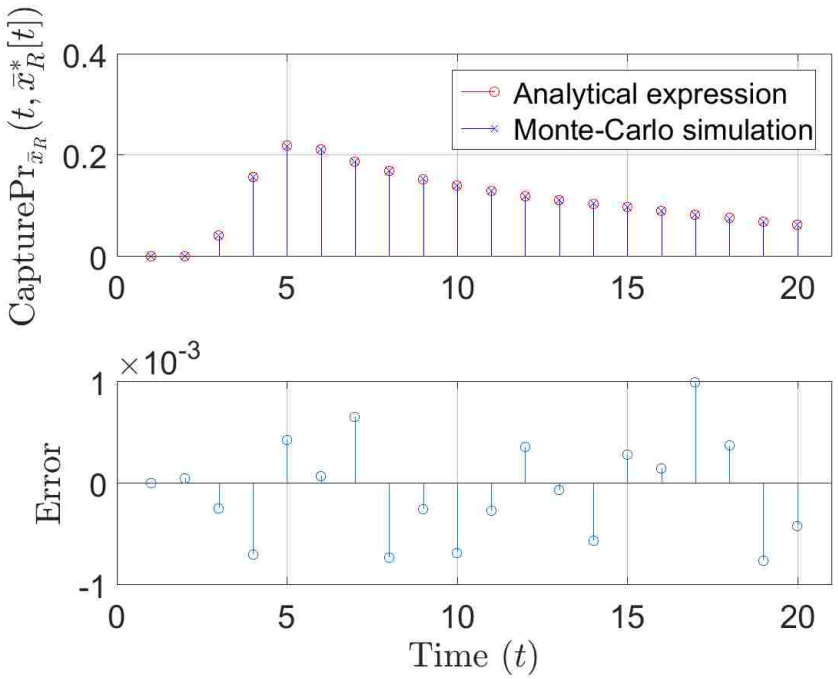


Figure 3.3: Solution to Problem 3.51 for robot G dynamics in (3.43), and validation of $\text{CapturePr}_{\bar{x}_R}(\tau, \bar{x}^*_R[\tau]; \bar{x}_G[0])$ via Monte-Carlo simulations. The optimal capture time is $\tau^* = 5$ and the likelihood of capture is $\text{CapturePr}_{\bar{x}_R}(\tau^*, \bar{x}^*_R[\tau^*]; \bar{x}_G[0]) = 0.219$.

The mean position of the robot G moves in a straight line $\mu_G[\tau]$, as it is the trajectory of (3.43a) when the input is always $\bar{\mu}^{\boldsymbol{v}}_G$. The optimal time of capture is $\tau^* = 5$, the optimal capture position is $\bar{x}^*_R[\tau^*] = [-1.8\ 0]^\top$, and the corresponding probability of robot R capturing robot G is 0.219. Note that at this instant, the reach set of the robot R does not cover the current mean position of the robot G, $\bar{\mu}[\tau^*] = [-1.7\ 0.3]^\top$. While the reach set covers the mean position of robot G at the next time instant $t = 6$, the uncertainty in (3.43) causes the probability of successful capture to further reduce. Counterintuitively, attempting to reach the mean $\mu_G[\tau]$ is not always best. Figure 3.3 shows the optimal capture probabilities obtained when solving Problem 3.51 for the dynamics (3.43).

3.6.3 Goal Robot with Double Integrator Dynamics

We now consider a more complicated capture problem, in which the disturbance is exponential (hence tracking the mean has little relevance because it is not the mode, the global maxima of the density), and the robot dynamics are more realistic. We solve Problem 3.49 for the system given by (3.44). Here, the disturbance set is $\mathcal{W} = \mathbb{R}^2_+$. Based on the mean of the stochastic acceleration $\boldsymbol{a}[t]$, the mean position of robot G has a parabolic trajectory due to the double integrator dynamics, as opposed to the linear trajectory seen in case of the point mass dynamics with Gaussian velocity model.

Proposition 8. *The characteristic function of the forward stochastic reach probability density of the robot G for dynamics* (3.44) *is*

$$\Psi_{\boldsymbol{x}_G}(\bar{\beta};\tau,\bar{x}_G[0]) = \exp(j\bar{\beta}^\top(A^\tau_{\mathrm{G,DI}}\bar{x}_G[0]))\prod_{t=0}^{\tau-1}\frac{\lambda_{\mathrm{ax}}\lambda_{\mathrm{ay}}}{(\lambda_{\mathrm{ax}}-j\bar{\alpha}_{2t})(\lambda_{\mathrm{ay}}-j\bar{\alpha}_{2t+1})} \tag{3.52}$$

where $\bar{\alpha} = \mathscr{C}^\top_{4\times(2\tau)}\bar{\beta} \in \mathbb{R}^{(2\tau)}$ *and* $\bar{\beta} \in \mathbb{R}^4$. *The forward stochastic reach probability density of the robot G is* $\psi_{\boldsymbol{x}_G}(\bar{x};\tau,\bar{x}_G[0]) = \mathscr{F}^{-1}\big\{\Psi_{\boldsymbol{x}_G}(\cdot;\tau,\bar{x}_G[0])\big\}(-\bar{x})$.

Proof: Apply (3.6) to the dynamics (3.44). ■

To solve Problem 3.49, we define $\mathrm{CapturePr}_{\bar{x}_R}(\cdot)$ as in (3.48). Since we are interested in just the position of robot G, we require only the marginal density of the forward stochastic reach probability density over the position subspace of robot G, $\psi^{\mathrm{pos}}_{\boldsymbol{x}_G}$. By Property P4, we have for $\bar{\gamma} = [\gamma_1\ \gamma_2] \in \mathbb{R}^2$,

$$\Psi^{\mathrm{pos}}_{\boldsymbol{x}_G}(\bar{\gamma};\tau,\bar{x}_G[0]) = \Psi_{\boldsymbol{x}_G}([\gamma_1\ 0\ \gamma_2\ 0]^\top;\tau,\bar{x}_G[0]). \tag{3.53}$$

Unlike the case with Gaussian disturbance, explicit expressions for the forward stochastic reach probability density $\psi_{\boldsymbol{x}_G}$ or its marginal density $\psi^{\mathrm{pos}}_{\boldsymbol{x}_G}$ are unavailable since the Fourier transform (3.52) is not standard.

We define a convex capture region $\text{CaptureSet}(\bar{y}_R) = \text{Box}(\bar{y}_R, a) \subseteq \mathbb{R}^2$ where $\bar{y}_R \in \mathbb{R}^2$ is the state of the robot R. We define $h(\bar{y}; \bar{y}_R, a) = \mathbf{1}_{\text{Box}(\bar{y}_R,a)}(\bar{y})$ as the indicator function corresponding to a 2-D box centered at $\bar{y}_R$ with edge length $2a > 0$ with $h(\bar{y}) = 1$ if $\bar{y} \in \text{CaptureSet}(\bar{y}_R)$ and zero otherwise. The Fourier transform of h is a product of *sinc* functions shifted by $\bar{y}_R$ (follows from Property P2 and [Bra86, Chapter 13])

$$H(\bar{\gamma}; \bar{y}_R, a) = \mathscr{F}\{h(\cdot; \bar{y}_R, a)\}(\bar{\gamma}) = 4a^2 \exp\left(-j\bar{y}_R^\top \bar{\gamma}\right) \frac{\sin(a\gamma_1)\sin(a\gamma_2)}{\gamma_1\gamma_2}. \tag{3.54}$$

Using the square-integrability of h and $\Psi^{\text{pos}}_{\boldsymbol{x}_G}$ (see [VHO17, Lem. 1 and 9]), we define $\text{CapturePr}_{\bar{x}_R}(\cdot)$ in (3.56). Equation (3.56) is evaluated using (3.52), (3.53), and (3.54). We use (3.56) as opposed (3.55) due to the unavailability of an explicit expression for $\psi^{\text{pos}}_{\boldsymbol{x}_G}$,

$$\begin{aligned} \text{CapturePr}_{\bar{x}_R}(\tau, \bar{x}_R[\tau]; \bar{x}_G[0]) &= \int_{\mathbb{R}^2} \psi^{\text{pos}}_{\boldsymbol{x}_G}(\bar{x}; \tau, \bar{x}_G[0]) h(\bar{x}; \bar{x}_R[\tau], a) d\bar{x} && (3.55) \\ &= \left(\frac{1}{2\pi}\right)^2 \int_{\mathbb{R}^2} \Psi^{\text{pos}}_{\boldsymbol{x}_G}(\bar{\gamma}; \tau, \bar{x}_G[0]) H(\bar{\gamma}; \bar{x}_R[\tau], a) d\bar{\gamma}. && (3.56) \end{aligned}$$

The numerical evaluation of the inverse Fourier transform of $\Psi^{\text{pos}}_{\boldsymbol{x}_G}$ to compute (3.55) will require two quadratures, resulting in a higher approximation error as compared to (3.56). See [VHO17] for experimental setup and implementation details.

We implement the problem with the following parameters: $T_s = 0.2$, $T = 9$, $a = 0.25$, $\lambda_{\text{ax}} = 0.25$, $\lambda_{\text{ay}} = 0.45$, $\bar{x}_G[0] = [1.5\ 0\ -0.5\ 2]^\top$, $\bar{x}_R[0] = [2.5\ 0]^\top$, and $\mathcal{U} = [-1.5, 1.5] \times [1, 4]$.

Figure 3.2 (right) shows the evolution of the mean position of the robot G and the optimal capture position for the robot R at time instants $1, 2, 3, 6$, and 9. For every $\tau \in [1, T]$, the contour plots of $\psi^{\text{pos}}_{\boldsymbol{x}_G}(\cdot; \tau, \bar{x}_G[0])$ were estimated via Monte-Carlo simulation since evaluating $\psi^{\text{pos}}_{\boldsymbol{x}_G}(\cdot; \tau, \bar{x}_G[0])$ via (2.12) over a grid is computationally

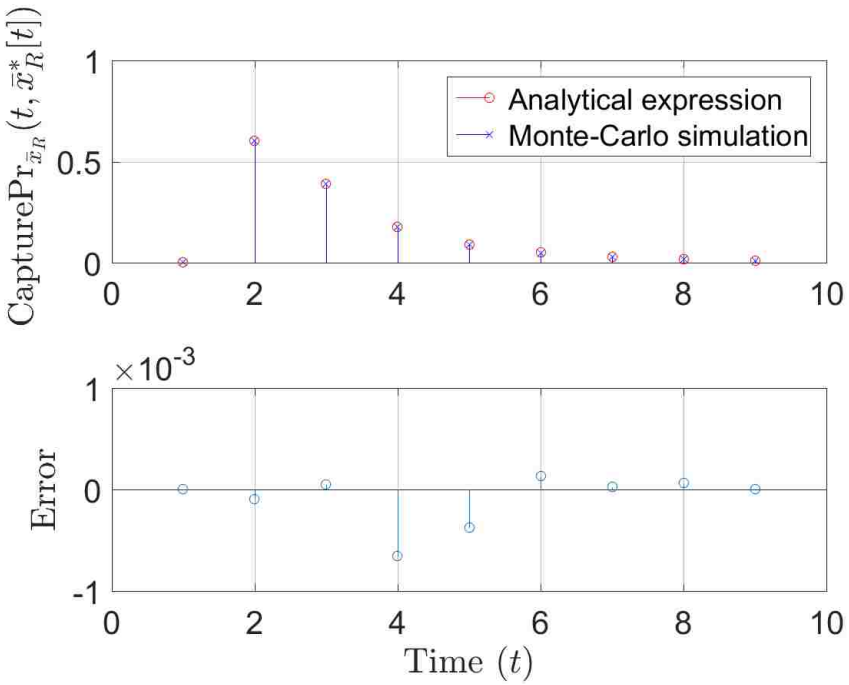


Figure 3.4: Solution to Problem 3.51 for robot G dynamics in (3.44), and validation of $\text{CapturePr}_{\bar{x}_R}(\tau, \bar{x}_R^*[\tau]; \bar{x}_G[0])$ via Monte-Carlo simulations. The optimal capture time is $\tau^* = 2$ and the capture probability is $\text{CapturePr}_{\bar{x}_R}(\tau^*, \bar{x}_R^*[\tau^*]; \bar{x}_G[0]) = 0.6044$.

expensive. Note that the mean position of the robot G does not coincide with the mode of $\psi^{\text{pos}}_{\boldsymbol{x}_G}(\cdot; \tau, \bar{x}_G[0])$ in contrast to the problem discussed for the Gaussian case. The optimal time of capture is at $\tau^* = 2$, the optimal capture position is $\bar{x}_R^*[\tau^*] = [1.9\ 0.55]^\top$, and the corresponding probability of robot R capturing robot G is 0.6044. Figure 3.4 shows the optimal capture probabilities obtained when solving Problem 3.51 for the dynamics (3.44), and the validation of the results.

3.6.4 Implementation Details

All computations were performed using MATLAB on an Intel Core i7 CPU with 3.4GHz clock rate and 16 GB RAM. We solved Problem 3.51 using MATLAB's built-in functions — `fmincon` for the optimization, `mvncdf` to compute the objective (3.48) for the Gaussian case, `integral` to compute the objective (3.56) for the exponential case, and `max` to compute the global optimum of Problem 3.49. In both the sections,

we used MPT for the reachable set calculation. All geometric computations were done in the facet representation. We computed the initial guess for the optimization of Problem 3.51 by performing Euclidean projection of the mean to the feasible set using CVX [BV04, Section 8.1.1]. Since computing the objective was costly, this operation saved significant computational time. The Monte-Carlo simulation used 5×10^5 particles. No offline computations were done in either of the cases.

The overall computation in the Gaussian case took 5.32 seconds for $T = 20$. Since the Gaussian case has explicit expressions for the forward stochastic reach probability density, the evaluation of the forward stochastic reach probability density for any given point $\bar{y} \in \mathcal{X}$ takes 1.6 milliseconds on average. For the exponential case, the overall computation took 488.55 seconds ($\sim$ 8 minutes) for $T = 9$. The numerical evaluation of the improper integral (3.56) is the major cause of increase in runtime. The evaluation of the forward stochastic reach probability density for any given point $\bar{y} \in \mathcal{X}$ using (2.12) takes about 10.5 seconds, and the runtime and the accuracy depend heavily on the point $\bar{y}$ as well as the bounds used for the integral approximation. However, the evaluation of $\text{CapturePr}_{\bar{x}_R}(\cdot)$ using (3.56) is much faster (0.81 seconds) because $H(\bar{\gamma}; \bar{y}_R, a)$ is a decaying, 2-D sinc function (decaying much faster than the characteristic function).

The decaying properties of the integrand in (3.56) and characteristic functions in general permits approximating the improper integrals in (2.12) and (3.56) by as a proper integral with suitably defined finite bounds. The tradeoff between accuracy and computational speed, common in quadrature techniques, dictates the choice of the bound. A detailed analysis of various quadrature techniques, their computational complexity, and their error analysis can be found in [Pre+07, Chapter 4].

3.6.5 Experimental Validation

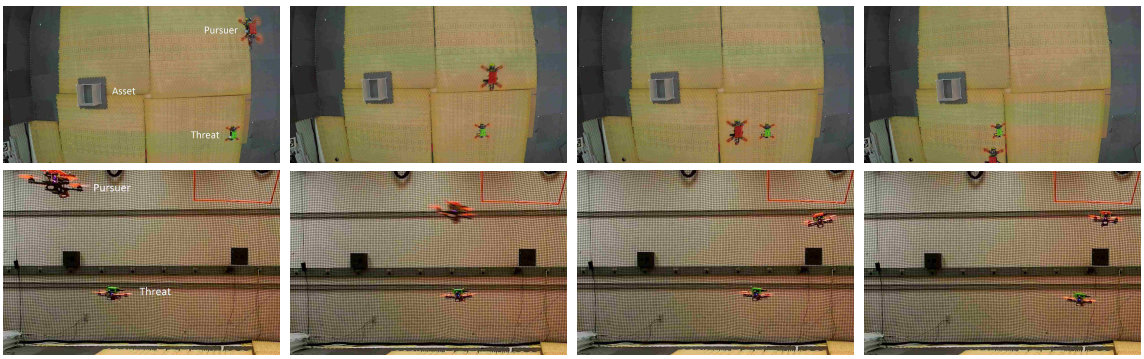
We extended this approach to the case of multiple pursuers in [Vin+18b]. In addition, we implemented this approach on a hardware testbed with quadrotors. The objective of this experiment was to demonstrate the utility of forward stochastic reachability to design autonomous pursuers to capture a human-controlled target UAV. We modeled the variation of the target trajectories resulting from the human-in-the-loop by a Gaussian disturbance added to an adversarial trajectory.

We modeled the target as well as the pursuer using 12-dimensional quadrotor dynamics, linearized about the hover state and controls. We closed the loop for the target using a LQR controller driving it towards the asset. To account for the mismatch between this controller and the human's actions, we added a Gaussian disturbance to the target dynamics. Next, we designed a receding horizon control for the pursuer using Problem 3.49. To enable faster computation, we approximated the objective in Problem 3.51 to a quadratic cost. See [Vin+18b] for more details.

Figures 3.5 and 3.6 show that the pursuer intercepts the threat despite the model mismatch (due to human controller), demonstrating the robustness provided by the receding horizon control framework. An update in the desired trajectory occurs on average every 0.33 seconds.

3.7 Application: Stochastic Motion Planning using α-Probabilistic Occupied Set and Successive Convexification

In this section, we utilize probabilistic occupancy function and α-probabilistic occupied sets to predict the keep-out regions in a stochastic motion planning problem to achieve desired probabilistic safety (Figure 1.1). We use successive convexification [MSA16; Mao+17] techniques to plan the trajectories around the characterized keep-out sets, in a receding horizon control framework.



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

Figure 3.5: Overhead and sideview snapshots of Experiment 1 with receding horizon control. (a) Start of the experiment (b) Pursuer moving towards the optimal location of intercept that was computed online (c) Pursuer at the optimal location of intercept (d) Successful intercept with $\boldsymbol{x}_G[\tau] \in \text{CatchSet}(\overline{x}_{P_i}[\tau])$. See Experiment 1 video at https://youtu.be/eFGg7U7gEQw.

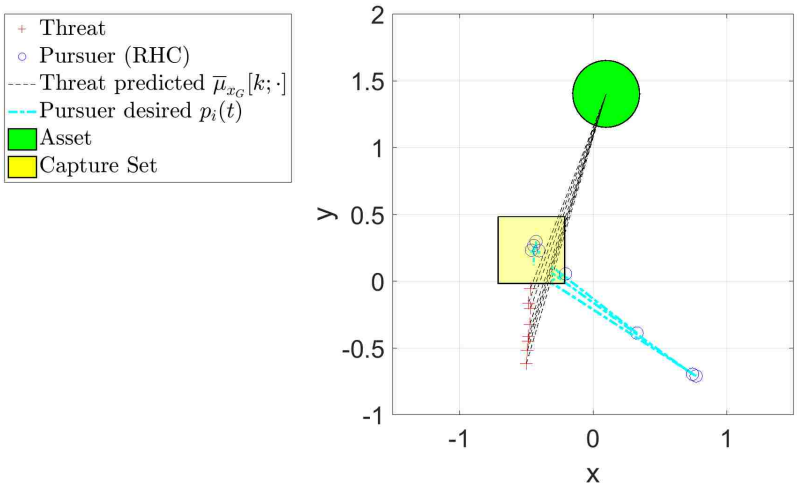


Figure 3.6: Predicted threat mean trajectory and the desired pursuer trajectory (a fitted polynomial $p_i(t)$) in Experiment 1. The pursuer demonstrates robustness by intercepting a threat that follows a path inconsistent with the predictions from the threat model.

3.7.1 Problem Setup

We consider the motion planning problem over the time interval $\mathbb{N}_{[0,T]}$ with a time horizon $T > 0$. The robot dynamics are assumed to be LTI,

$$\overline{x}_{\text{robot}}[t+1] = A_{\text{robot}}\overline{x}_{\text{robot}}[t] + B_{\text{robot}}\overline{u}_{\text{robot}}[t] \tag{3.57}$$

with state $\overline{x}_{\text{robot}}[t] \in \mathcal{X} = \mathbb{R}^n$, input $\overline{u}_{\text{robot}}[t] \in \mathcal{U} \subset \mathbb{R}^m$ ($\mathcal{U}$ is compact), and known matrices $A_{\text{robot}}, B_{\text{robot}}$ and initial state $\overline{x}_{\text{robot}}[0]$. The environment is assumed to have N_{obs} obstacles, and their dynamics are also assumed to be LTI ($j \in \mathbb{N}_{[1,N_{\text{obs}}]}$)

$$\boldsymbol{x}_{\text{obs},j}[t+1] = A_{\text{obs},j}\boldsymbol{x}_{\text{obs},j}[t] + F_{\text{obs},j}\boldsymbol{w}_{\text{obs},j}[t], \tag{3.58}$$

with state $\boldsymbol{x}_{\text{obs},j}[t] \in \mathcal{X}$, disturbance $\boldsymbol{w}_{\text{obs},j}[t] \in \mathbb{R}^p$ that has a known probability measure $\mathbb{P}_{\boldsymbol{w}_{\text{obs},j}}$, and known matrices $A_{\text{obs},j}, F_{\text{obs},j}$ and initial state $\overline{x}_{\text{obs},j}[0]$. We use $\boldsymbol{X}_{\text{obs}}[t]$ to describe the obstacle configuration $\boldsymbol{X}_{\text{obs}}[t] = [\boldsymbol{x}_{\text{obs},1}^\top[t]\ \boldsymbol{x}_{\text{obs},2}^\top[t]\ \ldots\ \boldsymbol{x}_{\text{obs},N_{\text{obs}}}^\top[t]]^\top$, with associated probability measure $\mathbb{P}^{t,\overline{X}_{\text{obs}}[0]}_{\boldsymbol{X}_{\text{obs}}}$ parameterized by time t and the initial obstacle configuration $\overline{X}_{\text{obs}}[0] \in \mathcal{X}^{N_{\text{obs}}}$.

Under Assumptions 1 and 2, the rigid body robot is characterized by its state $\overline{x}_{\text{robot}}[t]$ and a compact rigid body shape $\mathcal{R}(\overline{x}_{\text{robot}}[t]) = \{\overline{x}_{\text{robot}}[t]\} \oplus \mathcal{R}(\overline{0}) \subset \mathcal{X}$. Similarly, each of the rigid body obstacles is characterized by their respective state $\boldsymbol{x}_{\text{obs},j}[t]$ and a compact rigid body shape $\mathcal{O}_j(\boldsymbol{x}_{\text{obs},j}[t]) = \{\boldsymbol{x}_{\text{obs},j}[t]\} \oplus \mathcal{O}_j(\overline{0}) \subset \mathcal{X}$ with $j \in \mathbb{N}_{[1,N_{\text{obs}}]}$. Under Assumption 2, the collision avoidance problem can be equivalently reformulated as requiring the robot, now modeled as a point mass at $\overline{x}_{\text{robot}}[t]$, avoid obstacles with the rigid body shape $\mathcal{O}_j^+(\overline{0}) = \mathcal{O}_j(\overline{0}) \oplus (-\mathcal{R}(\overline{0}))$.

Since $\mathcal{R}(\overline{0})$ and $\mathcal{O}_j(\overline{0})$ are compact, the set $\mathcal{O}_j^+(\overline{0})$ is compact [Web94, Thm. 1.8.10(v)], and therefore bounded. The boundedness of $\mathcal{O}_j^+(\overline{0})$ allows us to specify a separation distance $r_j > 0$ between the states $\boldsymbol{x}_{\text{obs},j}[t]$ and $\overline{x}_{\text{robot}}[t]$ which guar-

antees collision avoidance. We define the collection of these separation distances r_j as $\overline{r} = [r_1\ r_2\ \ldots\ r_{N_{\text{obs}}}]^\top \in \mathbb{R}^{N_{\text{obs}}}_{>0}$, the *safe separation vector*. We can conservatively approximate the original collision avoidance problem by requiring the point mass describing the robot to stay out of a union of balls, $\cup_{j=1}^{N_{\text{obs}}} \text{Ball}(\boldsymbol{x}_{\text{obs},j}[t], r_j)$.

For this problem, we characterize a probabilistic occupancy function for the obstacle configuration, $\phi_{\boldsymbol{X}_{\text{obs}}} : \mathcal{X} \to [0,1]$,

$$\phi_{\boldsymbol{X}_{\text{obs}}}(\overline{y}; t, \overline{X}_{\text{obs}}[0], \overline{r}) = \mathbb{P}^{t,\overline{X}_{\text{obs}}[0]}_{\boldsymbol{X}_{\text{obs}}} \left\{ \cup_{j=1}^{N_{\text{obs}}} \{\boldsymbol{x}_{\text{obs},j}[t] \in \{\overline{z} \in \mathcal{X} : \overline{y} \in \text{Ball}(\overline{z}, r_j)\}\} \right\} \tag{3.59}$$

The function $\phi_{\boldsymbol{X}_{\text{obs}}}(\overline{y}; t, \overline{X}_{\text{obs}}[0], \overline{r})$ provides the probability that $\cup_{j=1}^{N_{\text{obs}}} \text{Ball}(\boldsymbol{x}_{\text{obs},j}[t], r_j)$ occupies the state of interest $\overline{y} \in \mathcal{X}$ at time t, given the safe separation vector $\overline{r}$, and initial obstacle configuration $\overline{X}_{\text{obs}}[0]$.

The stochastic motion planning problem of interest may be formulated as (3.60).

$$\begin{aligned}
\underset{\substack{\overline{x}_{\text{robot}}[1],\ldots,\overline{x}_{\text{robot}}[T],\\ \overline{u}_{\text{robot}}[0],\ldots,\overline{u}_{\text{robot}}[T-1]}}{\text{minimize}} \quad & J\big(\overline{u}_{\text{robot}}[0], \ldots, \overline{u}_{\text{robot}}[T-1], \overline{x}_{\text{robot}}[1], \ldots, \overline{x}_{\text{robot}}[T]\big) \\
\text{subject to} \quad & \overline{x}_{\text{robot}}[t+1] = A_{\text{robot}}\overline{x}_{\text{robot}}[t] + B_{\text{robot}}\overline{u}_{\text{robot}}[t], \quad \forall t \in \mathbb{N}_{[0,T-1]} \\
& \overline{u}_{\text{robot}}[t] \in \mathcal{U}, \quad \forall t \in \mathbb{N}_{[0,T-1]} \\
& \overline{x}_{\text{robot}}[t] \in \text{SafeSet}, \quad \forall t \in \mathbb{N}_{[0,T-1]} \\
& \overline{x}_{\text{robot}}[T] \in \text{GoalSet} \\
& \phi_{\boldsymbol{X}_{\text{obs}}}(\overline{x}_{\text{robot}}[t]; t, \overline{X}_{\text{obs}}[0], \overline{r}) < \alpha, \quad \forall t \in \mathbb{N}_{[1,T]}
\end{aligned} \tag{3.60}$$

We seek to minimize a convex cost function $J : \mathcal{U}^T \times \mathcal{X}^T \to \mathbb{R}$ while assuring that 1) the robot stays within a convex and compact safe set $\text{SafeSet} \subseteq \mathcal{X}$ at all times, 2) the robot reaches a convex and compact goal set $\text{GoalSet} \subseteq \text{SafeSet}$ at time T, and 3) the probability of collision of the robot with any of the rigid-body obstacles is below a maximum acceptable probability of collision $\alpha \in (0,1]$ for each instant. We know the

initial obstacle configuration $\overline{X}_{\text{obs}}[0]$, the initial robot state $\overline{x}_{\text{robot}}[0]$, the dynamics of the robot (3.57) and the obstacles (3.58), and the safe separation vector $\overline{r} \in \mathbb{R}^{N_{\text{obs}}}_{>0}$.

3.7.2 Receding Horizon Control Framework

The uncertainty in the stochastic obstacle dynamics grows over time, resulting in large keep-out sets. Conservative overapproximations of these large sets may induce infeasibility, even when the original problem is feasible. Replanning the robot trajectory based on the information available about the obstacle movement can reduce this conservativism. This motivates the use of receding horizon control. We choose a control horizon $0 < N < T$ and solve a problem similar to (3.60), referred to as the *receding horizon optimal control problem*, for the time interval $\mathbb{N}_{[t,t+N]}$.

Assumption 3. *Full-state information is available about all the obstacles after executing the first action prescribed by the solution to the receding horizon optimal control problem.*

To utilize a receding horizon control framework, we have to solve an additional convex optimization problem to find $\overline{x}_{\text{robot}}[N]$ closest to the GoalSet,

$$\begin{aligned} \underset{\overline{y}}{\text{minimize}} \quad & \|\overline{x}_{\text{robot}}[N] - \overline{y}\|_2 \\ \text{subject to} \quad & \overline{y} \in \text{GoalSet} \end{aligned} \tag{3.61}$$

The optimal value of (3.61) is zero only when $\overline{x}_{\text{robot}}[N] \in \text{GoalSet}$, in which case the optimal solution is $\overline{y}^* = \overline{x}_{\text{robot}}[N]$. Using the fixed-risk approach in [BHW06], we replace the constraint $\phi_{\boldsymbol{X}_{\text{obs}}}(\overline{x}_{\text{robot}}[t]; t, \overline{X}_{\text{obs}}[0], \overline{r}) < \alpha$ for every $t \in \mathbb{N}_{[1,T]}$ in (3.60) with

$$\overline{x}_{\text{robot}}[t] \notin \bigcup_{j=1}^{N_{\text{obs}}} \text{PrOccupySet}_{\boldsymbol{x}_{\text{obs},j}} \left(\frac{\alpha}{N_{\text{obs}}}; t, \overline{x}_{\text{obs},j}[0], r_j \right). \tag{3.62}$$

We obtain the receding horizon optimal control problem in (3.63) by introducing the parameter $\lambda \geq 0$ and scalarizing the bi-criterion optimization problem (minimizing cost J as well as $\|\overline{x}_{\text{robot}}[N] - \overline{y}\|_2$ (see [BV04, Sec. 4.7.4]),

$$
\begin{aligned}
\underset{\substack{\overline{y},\overline{x}_{\text{robot}}[1],\ldots,\overline{x}_{\text{robot}}[N],\\ \overline{u}_{\text{robot}}[0],\ldots,\overline{u}_{\text{robot}}[N-1]}}{\text{minimize}} \quad & J\big(\overline{u}_{\text{robot}}[0],\ldots,\overline{u}_{\text{robot}}[N-1],\overline{x}_{\text{robot}}[1],\ldots,\overline{x}_{\text{robot}}[N]\big) \\
& \qquad +\lambda\|\overline{x}_{\text{robot}}[N] - \overline{y}\|_2 \\
\text{subject to} \quad & \overline{x}_{\text{robot}}[t+1] = A_{\text{robot}}\overline{x}_{\text{robot}}[t] + B_{\text{robot}}\overline{u}_{\text{robot}}[t], \quad \forall t \in \mathbb{N}_{[0,T-1]} \\
& \overline{u}_{\text{robot}}[t] \in \mathcal{U}, \quad \forall t \in \mathbb{N}_{[0,T-1]} \\
& \overline{x}_{\text{robot}}[t] \in \text{SafeSet}, \quad \forall t \in \mathbb{N}_{[0,T-1]} \\
& \overline{y} \in \text{GoalSet} \\
& \overline{x}_{\text{robot}}[t] \notin \textstyle\bigcup_{j=1}^{N_{\text{obs}}} \text{PrOccupySet}_{\boldsymbol{x}_{\text{obs},j}}\left(\frac{\alpha}{N_{\text{obs}}}; t, \overline{x}_{\text{obs},j}[0], r_j\right), \\
& \qquad\qquad \forall t \in \mathbb{N}_{[1,T]}
\end{aligned}
\tag{3.63}
$$

where $J : \mathcal{X}^N \times \mathcal{U}^N \to \mathbb{R}$ approximates the original cost J (defined for the time horizon T) over the planning horizon N. The objective in (3.63) is convex with respect to the decision variables $\overline{x}_{\text{robot}}[\cdot]$ and $\overline{u}_{\text{robot}}[\cdot]$. We can interpret λ in (3.63) as a way to emphasize the relative importance of being close to the goal set with respect to the optimization of the cost function J. For large λ, the solver attempts to generate a trajectory that will minimize the distance of the terminal state at the expense of a potential increase in the cost function J.

3.7.3 Numerical Simulation

For brevity, we omit the details of the motion planner, successive convexification [Mao+17; MSA16], used to solve (3.63). The details of this technique, its adaptation to solve (3.63), and the analysis of the computational effort can be found in [Vin+18a].

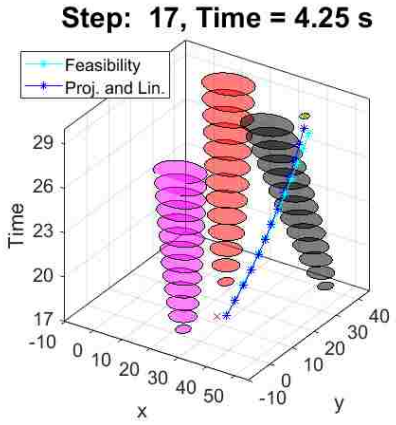


Figure 3.7: Planning over a single control horizon with the uncertainty in the obstacle location growing over time. The sets shown are the α-probabilistic occupied sets for each of the obstacle at each time step in the future. We compare the paths different two stages of the successive convexification planning (the feasible trajectory generation step and the project-and-linearize-based trajectory optimization).

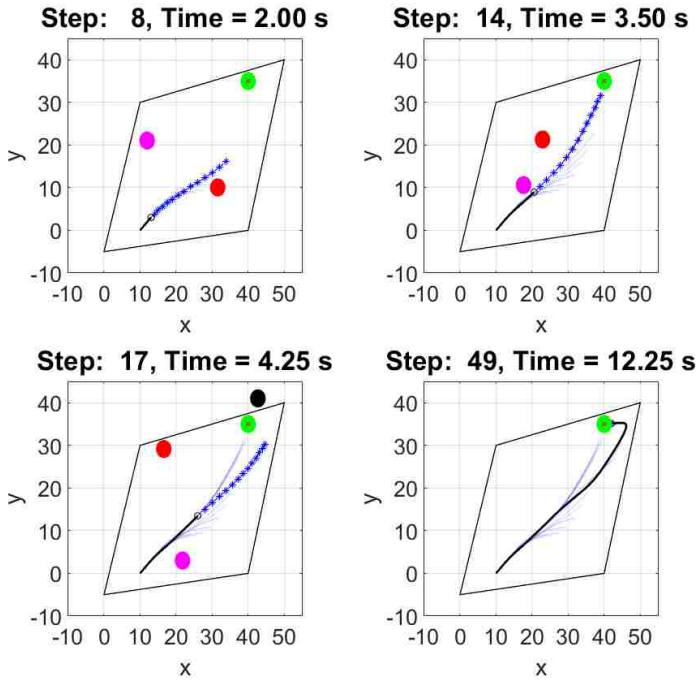


Figure 3.8: Receding horizon control-based solution to the stochastic motion planning problem. The motion planning was done using successive convexification [Mao+17; MSA16] with obstacle predictions provided probabilistic occupancy functions. The robot must stay within the quadrilateral, eventually reach the green region, and avoid stochastically moving obstacles (red, pink, and black). The faded blue lines and the black line show the intermediate plans and the executed trajectory respectively.

We consider the planning problem with double integrator robot dynamics (3.57),

$$A_{\text{robot}} = \begin{bmatrix} 1 & 0 & T_\Delta & 0 \\ 0 & 1 & 0 & T_\Delta \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}, \& \; B_{\text{robot}} = \begin{bmatrix} \frac{T_\Delta^2}{2} & 0 \\ 0 & \frac{T_\Delta^2}{2} \\ T_\Delta & 0 \\ 0 & T_\Delta \end{bmatrix}.$$

with sampling time $T_\Delta = 0.25$ s, the initial robot location $\overline{x}_{\text{robot}}[0] = [10\ 0]^\top$, and $\mathcal{U} = [-2, 2]^2$. We set $N_{\text{obs}} = 3$, and presume double integrator dynamics for the obstacles (3.58) with $A_{\text{obs},j} = A_{\text{robot}}$, $F_{\text{obs},j} = B_{\text{robot}}$, and Gaussian $\boldsymbol{w}_{\text{obs},j}[t] \sim \mathcal{N}(\overline{\mu}_{\boldsymbol{w}_{\text{obs},j}}, \Sigma_{\boldsymbol{w}_{\text{obs},j}})$, $j \in \mathbb{N}_{[1,N_{\text{obs}}]}$. We choose the obstacle initial positions $\overline{x}_{\text{obs},1}[0] = [10\ 25]^\top$, $\overline{x}_{\text{obs},2}[0] = [35\ 5]^\top$, $\overline{x}_{\text{obs},3}[0] = [70\ 80]^\top$, the mean values $\overline{\mu}_{\boldsymbol{w}_{\text{obs},1}} = [3\ -3]^\top$, $\overline{\mu}_{\boldsymbol{w}_{\text{obs},2}} = [-2\ 3]^\top$, $\overline{\mu}_{\boldsymbol{w}_{\text{obs},3}} = [-3\ -5]^\top$, and the covariance matrix $\Sigma_{\boldsymbol{w}_{\text{obs},j}} = \begin{bmatrix} 2.60 & 0.09 \\ 0.09 & 0.58 \end{bmatrix}$ $\forall j \in \{1, 2, 3\}$. We solve (3.60) with a time horizon $T = 15$ s (60 time steps), the cost function $J(\cdot) = \sum_{t=0}^{T-1} \|\overline{u}_{\text{robot}}[t]\|_2$ to minimize as the control effort, SafeSet $= \text{conv}((50, 40), (40, 0), (0, -5), (10, 30))$ (see (2.4)), GoalSet $= \text{Ball}([40\ 35]^\top, 2)$, control horizon $N = 13$, maximum acceptable collision probability $\alpha = 0.001$, scalarization parameter $\lambda = 10^4$, and safe separation $\overline{r} = [2\ 2\ 2]$.

Figure 3.7 shows the evolution of $\mathcal{E}(\overline{\mu}_{\boldsymbol{x}_{\text{obs},j}}[t], Q^+_{\boldsymbol{x}_{\text{obs},j}}[t])$ via Algorithm 4 (with an ellipsoid fit [Vin+18a, Alg. 2])) for a single control horizon, for both the initial feasible solution and project-and-linearize solutions. Figure 3.8 shows the realization of the stochastic motion planning problem at different time instants. We simulated the obstacle motion by drawing samples from their respective disturbance probability densities. The probability of an obstacle (black) reaching the GoalSet becomes non-trivial at $t = 4.25s$, causing the planner to re-plan its trajectory.

In our simulations, we were able to plan the trajectory with a mean solve time of 0.33 seconds and standard deviation of 0.17 seconds.

3.8 Summary

This chapter discussed a grid-free and scalable framework for addressing the problem of forward stochastic reachability for linear systems. We also characterized sufficient conditions for the log-concavity of the forward stochastic reach probability measure and convexity of the forward stochastic reach set. We also utilized this analysis to define the probabilistic occupancy function and the α-probabilistic occupied set for rigid body obstacles with linear dynamics. Additionally, we characterized sufficient conditions for the log-concavity and the upper semi-continuity of the probabilistic occupancy function. Using these results, we characterized sufficient conditions for the convexity, closedness, and compactness of the α-probabilistic occupied set. We proposed two computationally efficient algorithms to compute the approximation of the α-probabilistic occupied set. We showed the benefit of the convexity analysis by considering two application problems — capture of a stochastic target and stochastic motion planning problems.

Chapter 4

Stochastic Reachability of a Target Tube: Theory and Control

4.1 Introduction

The problem of stochastic reachability of a target tube is motivated by the question: *what initial states of a stochastic dynamical system can be driven to stay within a target tube (a collection of time-stamped target sets) with a desired likelihood, while respecting the given bounds on control authority?* Figure 1.5 provides an illustration of this problem. This chapter discusses novel theory for this problem. Specifically, we propose sufficient conditions under which the optimal value functions are Borel measurable, upper semi-continuous, and log-concave, and the stochastic reach set is closed, bounded, compact, and convex. Using these convexity and compactness properties, we describe an underapproximative interpolation technique for the stochastic reach sets. We also consider the problem of synthesis open-loop and affine feedback controllers to maximize probabilistic safety. We discuss how these point-based stochastic reachability evaluations can provide underapproximations to the maximal reach probability. The theoretical results presented in this chapter enable the design of scalable, grid-free, and anytime algorithms to verify high-dimensional systems, discussed in Chapter 5.

4.2 Related Work

The problem of backward stochastic reachability has received significant attention in the verification literature. The problem of stochastic reachability of target tube, discussed in this chapter, subsumes existing work on stochastic viability and terminal hitting-time stochastic reach-avoid problems [Aba+08; SL10]. Stochastic viability problems are concerned with maximizing the probability that system stays within a time-invariant safe set for a given time horizon (target tube with time-invariant safe sets) [Aba+08]. Terminal hitting-time stochastic reach-avoid problems are concerned with maximizing the probability that system stays within a time-invariant safe set within the time horizon and hits a (potentially different) target set at the time horizon [SL10].

A dynamic programming formulation for the stochastic reachability problem for the general class of discrete-time stochastic hybrid systems was proposed in [Aba+07; Aba+08; SL10]. This formulation, based on Markov decision process theory [BS78], casts the stochastic reachability problem as a discrete-time stochastic optimal control problem. The dynamic programming approach yields optimal value functions which map the states to their maximal reach probability. The superlevel sets of these functions, the *stochastic reach sets*, are the sets of "good" initial states, i.e., the set of initial states from which the system may be driven to stay within the target tube with a probability greater than a given threshold. For stochastic reach-avoid problems, sufficient conditions have been proposed for the well-posedness of the stochastic reach-avoid problem and the existence of an optimal Markov policy [Din+13; Kar+17; Yan18; VO17]. However, little is known about the sufficient conditions that guarantee convexity and compactness of stochastic reach sets, which we address in this chapter.

4.3 Stochastic Reachability of a Target Tube

We define the target tube as $\mathscr{T} = \{\mathcal{T}_k\}_{k=0}^{N},\ \mathcal{T}_k \in \mathscr{B}(\mathcal{X})$. These are pre-determined time-stamped sets of states that are deemed safe at each time instant within the time horizon. Define the *reach probability of a target tube*, $r_{\overline{x}_0}^{\pi}(\mathscr{T})$, for known initial state $\overline{x}_0$ and a Markov policy π (see Section 2.5), as the probability that the execution with policy π lies within the target tube $\mathscr{T}$ for the entire time horizon. Formally, we define $r_{\overline{x}_0}^{\pi}(\mathscr{T})$ as (see [VO18c] for more details),

$$r_{\overline{x}_0}^{\pi}(\mathscr{T}) = \mathbb{P}_{\boldsymbol{X}}^{\overline{x}_0,\pi}\left\{\forall k \in \mathbb{N}_{[0,N]},\ \boldsymbol{x}_k \in \mathcal{T}_k\right\}. \tag{4.1}$$

Note that this definition of reach probability is consistent with the terminal hitting-time reach probability defined in [SL10] and the viability probability defined in [Aba+08] for appropriately defined target tubes. For brevity, we will refer to the event $\{\forall k \in \mathbb{N}_{[0,N]},\ \boldsymbol{x}_k \in \mathcal{T}_k\}$ as $\{\boldsymbol{X} \in \mathscr{T}\}$, with a slight abuse of notation.

Motivated by [SL10, Def. 10], we define a Markov policy π^* as a *maximal reach policy* when it is the optimal solution of (4.2),

$$r_{\overline{x}_0}^{\pi^*}(\mathscr{T}) = \sup_{\pi \in \mathcal{M}} r_{\overline{x}_0}^{\pi}(\mathscr{T}). \tag{4.2}$$

The solution of (4.2) may be characterized via dynamic programming, a straighforward extension of stochastic reachability [SL10, Thm. 11] and viability [Aba+08, Thm. 2]. Define $V_k^* : \mathcal{X} \to [0,1],\ k \in \mathbb{N}_{[0,N]}$, by the backward recursion for $\overline{x} \in \mathcal{X}$,

$$V_N^*(\overline{x}) = 1_{\mathcal{T}_N}(\overline{x}) \tag{4.3a}$$

$$V_k^*(\overline{x}) = \sup_{\overline{u} \in \mathcal{U}} 1_{\mathcal{T}_k}(\overline{x}) \int_{\mathcal{X}} V_{k+1}^*(\overline{y}) Q_k(d\overline{y}|\overline{x},\overline{u}). \tag{4.3b}$$

Then, the optimal value to (4.2) is $r_{\overline{x}_0}^{\pi^*}(\mathscr{T}) = V_0^*(\overline{x}_0)$ for every $\overline{x}_0 \in \mathcal{X}$. Here, $Q_k(\cdot|\overline{x},\overline{u})$ is given by (2.20). In some special cases, it has an explicit expression in terms of the

disturbance PDF (2.22).

The optimal value function $V_0^*(\overline{x}_0)$ assigns to each initial state $\overline{x}_0 \in \mathcal{X}$ the maximal reach probability for the given target tube, and these maps are not probability density functions themselves (they don't integrate to 1 over $\mathcal{X}$). By construction,

$$0 \leq V_k^*(\overline{x}) \leq 1_{\mathcal{T}_k}(\overline{x}), \quad \forall \overline{x} \in \mathcal{X}. \tag{4.4}$$

For $\alpha \in [0,1]$, we define the superlevel sets of $V_k^*(\cdot)$ as,

$$\mathcal{L}_k^{\pi^*}(\alpha, \mathscr{T}) = \{\overline{x} \in \mathcal{X} : V_k^*(\overline{x}) \geq \alpha\}. \tag{4.5}$$

Of special interest is the superlevel set of $V_0^*(\cdot)$, the α-level stochastic reach set,

$$\mathcal{L}_0^{\pi^*}(\alpha, \mathscr{T}) = \{\overline{x} \in \mathcal{X} : r_{\overline{x}_0}^{\pi^*}(\mathscr{T}) \geq \alpha\}. \tag{4.6}$$

Here, $\mathcal{L}_0^{\pi^*}(\alpha, \mathscr{T})$ is the set of states which satisfies the objective of staying within the given target tube with a probability greater than or equal to α. From (4.1), $\mathcal{L}^{\pi^*}(0, \mathscr{T}) = \mathcal{X}$.

Lemma 6. *If $\alpha > 0$, then $\mathcal{L}_k^{\pi^*}(\alpha, \mathscr{T}) \subseteq \mathcal{T}_k$, $\forall k \in \mathbb{N}_{[0,N]}$. Additionally, bounded $\mathcal{T}_k$ implies bounded $\mathcal{L}_k^{\pi^*}(\alpha, \mathscr{T})$ for any $k \in \mathbb{N}_{[0,N]}$ and $\alpha > 0$.*

Proof: For any $\overline{x} \in \mathcal{L}_k^{\pi^*}(\alpha, \mathscr{T})$, $V_k^*(\overline{x}) \geq \alpha$. By (4.4), we have $1_{\mathcal{T}_k}(\overline{x}) \geq V_k^*(\overline{x}) \geq \alpha > 0 \Rightarrow \overline{x} \in \mathcal{T}_k$. The boundedness of $\mathcal{L}_k^{\pi^*}(\alpha, \mathscr{T})$ follows by definition [Tao06b, Defn. 12.5.3]. ■

Figure 1.5 (page. 10) illustrated the definition of the target tube $\mathscr{T}$ and the stochastic reach set $\mathcal{L}_0^{\pi^*}(\alpha, \mathscr{T})$ (4.6). Problem (4.2) defines the problem of stochastic reachability of a target tube, and it subsumes existing work done on stochastic viability and stochastic reach-avoid problems [Aba+08; SL10; Aba+07]. For $\mathscr{T} = \{\mathcal{S}\}_{k=0}^N$, $r_{\overline{x}_0}^{\rho^*}(\mathscr{T})$ and $\mathcal{L}_0^{\pi^*}(\alpha, \mathscr{T})$ is the maximal probabilistic safety probability and maximally

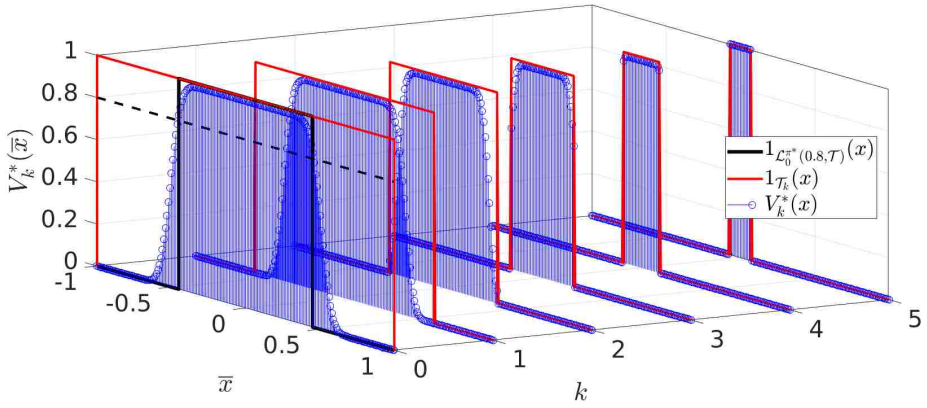


Figure 4.1: Dynamic programming (4.3) applied to example (4.7).

probabilistic safe set (stochastic viability set) respectively [Aba+08; Aba+07]. For $\mathscr{T} = \{\{\mathcal{S}\}_{k=0}^{N-1}, \mathcal{T}\}$, $r_{\overline{x}_0}^{\rho^*}(\mathscr{T})$ and $\mathcal{L}_0^{\pi^*}(\alpha, \mathscr{T})$ is the maximal terminal hitting-time reach-avoid probability and the terminal hitting-time stochastic reach-avoid set respectively [SL10].

Illustrative example: Consider the following one-dimensional system,

$$\boldsymbol{x}_{k+1} = \boldsymbol{x}_k + u_k + \boldsymbol{w}_k \tag{4.7}$$

with state $\boldsymbol{x}_k \in \mathbb{R}$, input $u_k \in [-1, 1]$, and disturbance $\boldsymbol{w}_k \sim \mathcal{N}(0, 0.001)$. We consider the stochastic reachability of a target tube $\mathscr{T} = \{[-\gamma^k, \gamma^k]\}_{k=0}^{N}$ with $\gamma = 0.6$ and time horizon $N = 5$. Using a step size of 0.01, the dynamic programming solution (4.3) is shown in Figure 4.1. As prescribed by (4.3a), we set $V_5^*(x) = 1_{\mathcal{T}_5}(x)$, and compute $V_k^*(\cdot)$ using the backward recursion (4.3b) over a grid of $\{-1, -0.99, \ldots, 0.99, 1\}$. The 0.8-level stochastic reach set is given by the superlevel set of $V_0^*(\cdot)$ at 0.8. From $V_k^*(\cdot)$ shown in Figure 4.1, we observe $0 \leq V_k^*(\overline{x}) \leq 1_{\mathcal{T}_k}(\overline{x}),\ \forall \overline{x} \in \mathcal{X}$ (4.4) and Lemma 6, $\mathcal{L}_k^{\pi^*}(\alpha, \mathscr{T}) \subseteq \mathcal{T}_k,\ \forall k \in \mathbb{N}_{[0,N]}$ and $\forall \alpha > 0$.

4.4 Problem Statements

Problem 3. *Provide sufficient conditions under which:*

1. *a maximal Markov policy to solve (4.2) exists,*
2. *$V_k^*(\cdot)$ is Borel-measurable, upper semi-continuous, and log-concave, and*
3. *the α-superlevel set of $V_0^*(\cdot)$, $\mathcal{L}_k^{\pi^*}(\alpha, \mathscr{T})$, is closed, compact, and convex,*

for every $k \in \mathbb{N}_{[0,N]}$ and $\alpha \in [0,1]$.

Problem 4. *Provide sufficient conditions under which an underapproximative interpolation of $\mathcal{L}_k^{\pi^*}(\alpha, \mathscr{T})$ can be constructed, given $\mathcal{L}_k^{\pi^*}(\alpha_1, \mathscr{T})$ and $\mathcal{L}_k^{\pi^*}(\alpha_2, \mathscr{T})$ with $\alpha \in [\alpha_1, \alpha_2]$ and $\alpha_1, \alpha_2 \in [0,1]$ for any $k \in \mathbb{N}_{[0,N]}$.*

Problem 5. *Demonstrate that the restriction of admissible policies for (4.2) to open-loop controllers yields an underapproximation $W_0^* : \mathcal{X} \to [0,1]$ to the maximal reach probability obtained via (4.2).*

Problem 5.a. *Characterize sufficient conditions under which:*

1. *the open-loop controller-based underapproximation to (4.2) is well-posed and convex,*
2. *the α-superlevel set of $W_0^*(\cdot)$, $\mathcal{K}_0^{\rho^*}(\alpha, \mathscr{T})$, is convex and compact for $\alpha \in (0,1]$, and*
3. *the underapproximative interpolation technique, described in Problem 4, holds for $\mathcal{K}_0^{\rho^*}(\cdot, \mathscr{T})$.*

Problem 6. *Construct an underapproximation of the maximal reach probability $r_{\overline{x}_0}^{\pi^*}(\mathscr{T})$ (4.2), based on the optimal reach probability attained by an unsaturated affine disturbance feedback controller (2.23).*

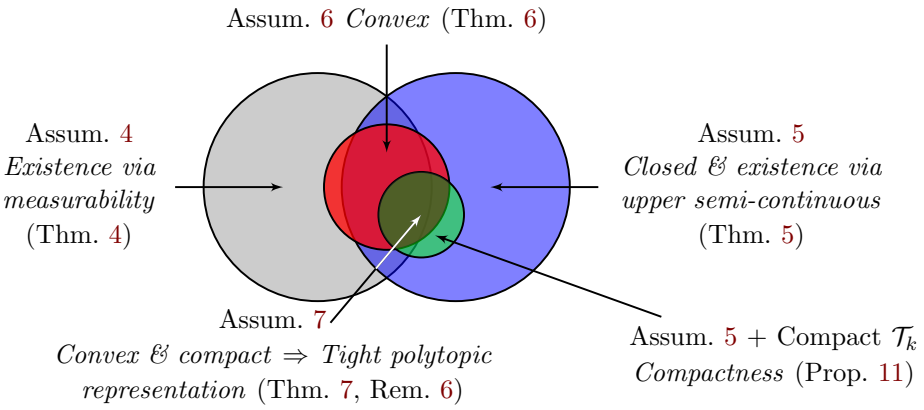


Figure 4.2: Various assumptions introduced in Section 4.5, and the resulting set properties (italicized) of the stochastic reach set $\mathcal{L}_k^{\pi^*}(\alpha, \mathscr{T})$.

Problem 6.a. *Formulate a chance constraint optimization problem to solve for the affine disturbance feedback controller that satisfies a chance constraint relaxation of the hard input constraints, up to a user-specified threshold.*

4.5 Properties of the Stochastic Reach Set

In this section, we will address Problem 3. We describe the relationship between various assumptions introduced in Section 4.5 in Figure 4.2.

4.5.1 Existence and Measurability: Borel Assumption

Sufficient conditions for the existence of an optimal Markov policy and the Borel-measurability of the optimal value functions have been formulated for reach-avoid problems [BS78, Sec. 8.3], [Din+13; Yan18; Kar+17; VO17]. These results impose continuity requirements on the stochastic kernel (Definition 2) and utilize a *measurable selection theorem* [HPV76, Thm. 2] to obtain the desired existence and measurability results. We now present straightforward extensions of these results to the more general problem of stochastic reachability of a target tube (4.2).

Definition 2. ***(Continuity of stochastic kernels)*** *Let $\mathcal{H}$ be the set of all bounded and Borel-measurable functions $h : \mathcal{X} \to \mathbb{R}$. A stochastic kernel $Q_k(\cdot|\overline{x}, \overline{u})$ is said to be:*

a. input-continuous*, if $\int_{\mathcal{X}} h(\overline{y})Q(d\overline{y}|\overline{x}, \overline{u})$ is continuous over $\mathcal{U}$ for each $\overline{x} \in \mathcal{X}$ for any $h \in \mathcal{H}$, and*

b. continuous*, if $\int_{\mathcal{X}} h(\overline{y})Q(d\overline{y}|\overline{x}, \overline{u})$ is continuous over $\mathcal{X} \times \mathcal{U}$ for any $h \in \mathcal{H}$.*

Recall that a function is said to be continuous if and only if its image of every sequence in its domain is also a convergent sequence [Tao06b, Thm. 13.4]. Since continuity over product spaces imply continuity over individual spaces [Tao06b, Lem. 13.2.1], continuous stochastic kernels are input-continuous. In other words, Definition 2b imposes a stronger requirement on the stochastic kernel $Q_k(\cdot|\overline{x}, \overline{u})$ than Definition 2a.

Assumption 4 (Borel)**.**

a. f_k is Borel-measurable over $\mathcal{X} \times \mathcal{U} \times \mathcal{W}$, $\forall k \in \mathbb{N}_{[0,N-1]}$,

b. $\mathcal{U}$ is compact,

c. $\mathscr{T} = \{\mathcal{T}_k\}_{k=0}^{N}$ such that $\mathcal{T}_k \subseteq \mathcal{X}$ are Borel $\forall k \in \mathbb{N}_{[0,N]}$, and

d. Q_k in (2.20) is input-continuous (Definition 2a).

In Theorem 4 and Proposition 9, we generalize the existence, measurability, and continuity results presented in [Kar+17, Props. 1 and 2] to the stochastic reachability problem of a target tube for a system described by a time-varying stochastic kernel. Note that unlike [Kar+17, Prop. 2], the structure in (4.2) permits exact characterization of where $V_k^*(\cdot)$ may be discontinuous in Proposition 9. Our proofs are similar in structure to [Kar+17, Prop. 1 and 2] [VO17, Thm. 1], and exploit the upper semi-continuous property of the objective of (4.3b) afforded by Definition 2a.

Theorem 4. *Under Assumption 4,*

a. $V_k^*(\cdot),\ \forall k \in \mathbb{N}_{[0,N]}$ *is Borel-measurable, and*

b. π^* *exists, and consists of Borel-measurable maps* $\mu_k^*(\cdot),\ \forall k \in \mathbb{N}_{[0,N-1]}$.

Proof: (By induction) (By induction) Since $\mathcal{T}_{N-1}, \mathcal{T}_N$ are Borel sets and indicator functions are bounded, $1_{\mathcal{T}_{N-1}}(\cdot)$ and $1_{\mathcal{T}_N}(\cdot)$ are bounded and Borel-measurable. The Borel-measurability and boundedness of $V_N^*(\cdot)$ follows from (4.3a).

Consider the base case $k = N-1$. Since $V_N^*(\cdot)$ is Borel-measurable (by above) and bounded (by (4.4)), $\int_{\mathcal{X}} V_N^*(\overline{y})Q(d\overline{y}|\overline{x},\overline{u})$ is continuous over $\mathcal{U}$ for each $\overline{x} \in \mathcal{X}$ by Definition 2a. Since continuity implies upper semi-continuity [BS78, Lem. 7.13 (b)] and $\mathcal{U}$ is compact, an optimal Borel-measurable input map $\mu_{N-1}^*(\cdot)$ exists and $\int_{\mathcal{X}} V_N^*(\overline{y})Q_N(d\overline{y}|\overline{x},\mu_{N-1}^*(\overline{x}))$ is Borel-measurable over $\mathcal{X}$ by [HPV76, Thm. 2]. Finally, $V_{N-1}^*(\cdot)$ is Borel-measurable since the product operator preserves Borel-measurability [Tao06b, Cor. 18.5.7].

For the case $k = t$ with $t \in \mathbb{N}_{[0,N-2]}$, assume for induction that $V_{t+1}^*(\cdot)$ is Borel-measurable. By the same arguments as above, a Borel-measurable $\mu_t^*(\cdot)$ exists and $V_t^*(\cdot)$ is Borel-measurable, completing the proof. ■

Since continuity implies upper semi-continuous, $\int_{\mathcal{X}} V_{k+1}^*(\overline{y})Q_k(d\overline{y}|\overline{x},\overline{u})$ is upper semi-continuous over $\mathcal{U}$ for every $\overline{x} \in \mathcal{X}$ and $k \in \mathbb{N}_{[0,N-1]}$. Thus, the set

$$\mathcal{U}_k(\overline{x},\lambda) = \left\{\overline{u} \in \mathcal{U} : \int_{\mathcal{X}} V_{k+1}^*(\overline{y})Q_k(\overline{y}|\overline{x},\overline{u})d\overline{y} \geq \lambda\right\}$$

is closed for every $\lambda \in \mathbb{R}$. Since $\mathcal{U}$ is compact (Assumption 4b) and $\mathcal{U}_k(\overline{x},\lambda)$ is closed, $\mathcal{U}_k(\overline{x},\lambda)$ is compact for every $\overline{x} \in \mathcal{X},\ k \in \mathbb{N}_{[0,N-1]}$, and $\lambda \in \mathbb{R}$ [Tao06b, Thm. 12.5.10a]. The compactness of $\mathcal{U}_k(\overline{x},\lambda)$ for every $\overline{x} \in \mathcal{X},\ k \in \mathbb{N}_{[0,N-1]}$, and $\lambda \in \mathbb{R}$ is another well-known sufficient condition for the existence of Markov policy (see [Aba+08, Thm. 1] [SL10, Thm. 11] [BS78, Lem. 3.1]).

Proposition 9. *Under Assumption 4, if Q_k is continuous, then*

a. $\int_{\mathcal{X}} V_{k+1}^*(\overline{y}) Q_k(d\overline{y}|\overline{x}, \mu_k^*(\overline{x}))$, $\forall k \in \mathbb{N}_{[0,N-1]}$ *is continuous over* $\mathcal{X}$, *and*

b. $V_k^*(\cdot)$, $\forall k \in \mathbb{N}_{[0,N-1]}$ *is piecewise-continuous over* $\mathcal{X}$ *where the discontinuities, if any, is restricted to the relative boundary of the target sets* $\partial\mathcal{T}_k$.

Proof: *Proof of a)*: Since continuous stochastic kernels are input-continuous, we have for every $k \in \mathbb{N}_{[0,N]}$, $V_k^*(\cdot)$ is Borel-measurable by Theorem 4 and bounded by (4.4). By Definition 2b, $\int_{\mathcal{X}} V_{k+1}^*(\overline{y}) Q_k(d\overline{y}|\overline{x}, \overline{u})$ is continuous over $\mathcal{X} \times \mathcal{U}$ for every $k \in \mathbb{N}_{[0,N-1]}$. By (4.4) and [Tao06b, Prop. 19.2.6], $\int_{\mathcal{X}} V_{k+1}^*(\overline{y}) Q_k(d\overline{y}|\overline{x}, \overline{u})$ is bounded and nonnegative. By [BS78, Prop. 7.32], we know that $\int_{\mathcal{X}} V_{k+1}^*(\overline{y}) Q_k(d\overline{y}|\overline{x}, \mu_k^*(\overline{x}))$ is lower semi-continuous and upper semi-continuous over $\mathcal{X}$, implying its continuity.

Proof of b): For every $k \in \mathbb{N}_{[0,N]}$, every $\overline{x} \in \text{int}(\mathcal{T}_k)$, and any sequence $\overline{x}_i \xrightarrow{i\to\infty} \overline{x}$ where $\overline{x}_i \in \mathcal{X}$, there exists $i_0 \in \mathbb{N}$ such that $\forall i \geq i_0$, $\overline{x}_i \in \mathcal{T}_k$. This implies $1_{\mathcal{T}_k}(\overline{x}_i) = 1_{\mathcal{T}_k}(\overline{x}) = 1$, $\forall i \geq i_0$, implying the continuity of $1_{\mathcal{T}_k}(\cdot)$, $\forall k \in \mathbb{N}_{[0,N]}$ over $\text{int}(\mathcal{T}_k)$. Since multiplication of continuous functions are continuous [Tao06b, Cor. 13.2.3a], $V_k^*(\overline{x})$ is continuous over $\text{int}(\mathcal{T}_k)$ by (4.3b) and Proposition 9a. By construction, $V_k^*(\overline{x}) = 0$ for every $\overline{x} \in \text{int}(\mathcal{X} \setminus \mathcal{T}_k)$, implying $V_k^*(\overline{x})$ is trivially continuous over $\text{int}(\mathcal{X} \setminus \mathcal{T}_k)$. Hence, $V_k^*(\cdot)$ is piecewise-continuous over $\mathcal{X}$, with discontinuities if any restricted to the relative boundary of $\mathcal{T}_k$. ■

By Proposition 9b, if for some $k \in \mathbb{N}_{[0,N-1]}$, the target set $\mathcal{T}_k = \mathcal{X}$, then $V_k^*(\cdot)$ is continuous over $\mathcal{X}$ for that particular k. For reachability problems that do not have safety constraints at $k = 0$ ($\mathcal{T}_0 = \mathcal{X}$), $V_0^*(\cdot)$ is continuous over $\mathcal{X}$, presuming the restrictions specified in Assumption 4 and continuous Q_k.

Assumptions 4a, 4b, and 4c impose requirements on the stochastic reachability problem that are easy to ensure. Based on [BS78, Sec. 8.3], Lemma 7 provides a set of sufficient conditions that guarantees Assumption 4d.

Lemma 7. *Given an affine-perturbed nonlinear system (2.18) with $g_k(\cdot,\cdot)$ continuous in $\mathcal{U}$ for each $\overline{x} \in \mathcal{X}$ and $k \in \mathbb{N}_{[0,N-1]}$; if the disturbance PDF $\psi_{\boldsymbol{w},k}$ is continuous over $\mathcal{W}$, then the stochastic kernel Q_k defined by (2.22) is input-continuous.*

Lemma 7 applies to linear systems (2.19) as well [VO17, Lem. 2]. If g_k is continuous over $\mathcal{X} \times \mathcal{U}$ for each $k \in \mathbb{N}_{[0,N-1]}$, then we have continuous (as opposed to input-continuous) Q_k.

4.5.2 Existence and Compactness: Closed assumption

In this section, we consider Assumption 5 to provide an alternative set of sufficient conditions to guarantee existence of an optimal Markov policy to solve (4.2).

Assumption 5 (Closed)**.**

a. f_k *is continuous over* $\mathcal{X} \times \mathcal{U} \times \mathcal{W}$, $\forall k \in \mathbb{N}_{[0,N-1]}$,

b. $\mathcal{X}$ *is closed.*

c. $\mathcal{U}$ *is compact,*

d. $\mathscr{T} = \{\mathcal{T}_k\}_{k=0}^{N}$ *such that* $\mathcal{T}_k \subseteq \mathcal{X}$ *are closed* $\forall k \in \mathbb{N}_{[0,N]}$, *and*

The key difference between Assumptions 4 and 5 is the relaxation (replacement) of Assumption 4d, the continuity requirements on Q_k, with stricter requirements on f_k, $\mathcal{X}$, and $\mathscr{T}$. Note that Assumption 4 imposes restrictions on $\psi_{\boldsymbol{w},k}$ but not on $\mathscr{T}$, whereas Assumption 5 imposes restrictions on $\mathscr{T}$ but not on $\psi_{\boldsymbol{w},k}$. Hence, we do not expect either of these assumptions to subsume the other (see Figure 4.2).

For Assumption 5, Theorem 5 guarantees the existence of an optimal Markov policy and upper semi-continuous optimal value functions. In contrast to the proof of Theorem 4, the proof of Theorem 5 uses Proposition 10 to guarantee that the objective of (4.3b) is upper semi-continuous, and then uses [BS78, Prop. 7.33] to

guarantee that the optimal value functions are upper semi-continuous. Note that Proposition 10 does not impose any restrictions on the stochastic kernel Q_k.

Proposition 10. *Suppose Assumptions 5a and 5b holds. For every bounded, non-negative, and upper semi-continuous function* $h : \mathcal{X} \to \mathbb{R}$, $\int_{\mathcal{X}} h(\overline{y})Q_k(d\overline{y}|\overline{x},\overline{u})$, $\forall k \in \mathbb{N}_{[0,N-1]}$ *is upper semi-continuous over* $\mathcal{X} \times \mathcal{U}$.

Proof: By (2.21), we can rewrite $\int_{\mathcal{X}} h(\overline{y})Q_k(d\overline{y}|\overline{x},\overline{u})$ as $\int_{\mathcal{X}} h(f_k(\overline{x},\overline{u},\overline{w}))\psi_{\boldsymbol{w},k}(\overline{w})d\overline{w}$. Note that $h(f_k(\overline{x},\overline{u},\overline{w}))$ is upper semi-continuous over $\mathcal{X} \times \mathcal{U} \times \mathcal{W}$ by Assumption 5a and the fact that upper semi-continuous function of a continuous function is upper semi-continuous [RW09, Ex. 1.4], and the assumption that $h(\cdot)$ is upper semi-continuous. Additionally, $h(f_k(\overline{x},\overline{u},\overline{w}))$ is bounded and non-negative since $h(\cdot)$ is bounded and non-negative. If $L \in \mathbb{R}$ is an upper bound of $h(\cdot)$, then $L - h(f_k(\overline{x},\overline{u},\overline{w}))$ is non-negative and l.s.c over $\mathcal{X} \times \mathcal{U}$ for every $\overline{w} \in \mathcal{W}$. By Borel-measurability of h, $h(f_k(\overline{x},\overline{u},\boldsymbol{w}))$ is a non-negative random vector defined on $(h(\mathcal{X}), \mathscr{B}(h(\mathcal{X})))$. From Fatou's lemma [CT97, Sec. 6.2, Thm. 2.1] and the fact that $L - h(f_k(\overline{x},\overline{u},\overline{w}))$ is l.s.c, Borel-measurable, and non-negative, we have

$$\begin{aligned}\liminf_i \int_{\mathcal{X}} (L - h(f_k(\overline{x}_i,\overline{u}_i,\overline{w})))\psi_{\boldsymbol{w},k}(\overline{w})d\overline{w} \\ \geq \int_{\mathcal{X}} \liminf_i (L - h(f_k(\overline{x}_i,\overline{u}_i,\overline{w})))\psi_{\boldsymbol{w},k}(\overline{w})d\overline{w} \\ \geq \int_{\mathcal{X}} (L - h(f_k(\overline{x},\overline{u},\overline{w})))\psi_{\boldsymbol{w},k}(\overline{w})d\overline{w}. \end{aligned} \tag{4.8}$$

By the linearity properties of the Lebesgue integral on (4.8) [Tao06b, Prop. 19.2.6c],

$$\limsup_i \int_{\mathcal{X}} h(f_k(\overline{x}_i,\overline{u}_i,\overline{w}))\psi_{\boldsymbol{w},k}(\overline{w})d\overline{w} \leq \int_{\mathcal{X}} h(f_k(\overline{x},\overline{u},\overline{w}))\psi_{\boldsymbol{w},k}(\overline{w})d\overline{w}, \tag{4.9}$$

which completes the proof. ■

Theorem 5. *Under Assumption 5,*

a. $V_k^*(\cdot),\ \forall k \in \mathbb{N}_{[0,N]}$ *is upper semi-continuous over* $\mathcal{X}$*,*

b. π^* *exists, and it consists of Borel-measurable maps* $\mu_k^*(\cdot),\ \forall k \in \mathbb{N}_{[0,N-1]}$*, and*

c. $\mathcal{L}_k^{\pi^*}(\alpha, \mathscr{T}),\ \forall k \in \mathbb{N}_{[0,N]}, \forall \alpha \in [0,1]$ *is closed.*

Proof: Since $\mathcal{T}_k$ and $\mathcal{X}$ are closed, $1_{\mathcal{T}_k}(\cdot),\ \forall k \in \mathbb{N}_{[0,N]}$ is upper semi-continuous over $\mathcal{X}$. Hence, $V_N^*(\cdot)$ is upper semi-continuous over $\mathcal{X}$.

Consider the base case $k = N-1$. Due to closedness of $\mathcal{T}_N$, $V_N^*(\cdot)$ is upper semi-continuous, and $V_N^*(\cdot)$ is bounded and non-negative by (4.4). Hence, $\int_{\mathcal{X}} V_N^*(\overline{y}) Q_{N-1}(d\overline{y}|\overline{x}, \overline{u})$ is upper semi-continuous over $\mathcal{X} \times \mathcal{U}$ by Proposition 10. By a selection result for semi-continuous cost functions [BS78, Prop. 7.33] and compactness of $\mathcal{U}$, an optimal Borel-measurable input map $\mu_{N-1}^*(\cdot)$ exists and $\int_{\mathcal{X}} V_N^*(\overline{y}) Q_N(d\overline{y}|\overline{x}, \mu_{N-1}^*(\overline{x}))$ is upper semi-continuous over $\mathcal{X}$. Since upper semi-continuity is preserved under multiplication [Put05, Props. B.1], $V_{N-1}^*(\cdot)$ is upper semi-continuous over $\mathcal{X}$ by (4.3b).

For the case $k = t$ with $t \in \mathbb{N}_{[0,N-2]}$, assume for induction that $V_{t+1}^*(\cdot)$ is upper semi-continuous. By the same arguments as above, a Borel-measurable $\mu_t^*(\cdot)$ exists and $V_t^*(\cdot)$ is upper semi-continuous, completing the proof for a) and b).

Upper semi-continuity of $V_k^*(\cdot),\ \forall k \in \mathbb{N}_{[0,N]}$, implies that $\mathcal{L}_k^{\pi^*}(\alpha, \mathscr{T}),\ \forall k \in \mathbb{N}_{[0,N]}$ is closed for $\alpha \in [0,1]$. ■

Remark 4. *The stochastic reachability problem of a target tube* (4.2) *is well-posed under Assumptions 4 or 5.*

Proposition 11. *Under Assumption 5, if* $\mathcal{T}_k$ *is bounded (and thereby compact) for some* $k \in \mathbb{N}_{[0,N]}$*, then* $\mathcal{L}_k^{\pi^*}(\alpha, \mathscr{T}),\ \forall \alpha \in (0,1]$*, is compact.*

Proof: Follows from Heine-Borel theorem, and closedness (Theorem 5c) and boundedness (Lemma 6) of $\mathcal{L}_k^{\pi^*}(\alpha, \mathscr{T})$. ■

By Proposition 11, if $\mathcal{T}_0$ is bounded (and thereby compact), then $\mathcal{L}_0^{\pi^*}(\alpha, \mathscr{T})$, $\forall \alpha \in (0, 1]$ is compact, under Assumption 5. Further, if $\mathcal{X}$ is bounded, then $\mathcal{L}^{\pi^*}(0, \mathscr{T}) = \mathcal{X}$ is compact.

Relationship between Markov policies and general policies for stochastic reachability

We can also use the well-posedness of the reachability problem to characterize when the equivalence in performance of a Markov policy and a general policy (see Section 2.5.2 for their definitions).

Lemma 8. *Let Assumption 5 hold. For every $\overline{x}_0 \in \mathcal{T}_0$ and a general policy π', there exists a Markov policy $\pi \in \mathcal{M}$ such that*

$$r_{\overline{x}_0}^{\pi}(\mathscr{T}) = r_{\overline{x}_0}^{\pi'}(\mathscr{T}). \tag{4.10}$$

Proof: The hypothesis ensures that the stochastic reachability problem is well-posed (Remark 4). Further, the reachability problem meets the criteria imposed by the finite horizon stochastic optimal control model used in [BS78, Defn. 1]. We have (4.10) from [BS78, Prop. 8.1] and arguments similar to the proof of [Aba+08, Thm. 1] for every initial state $\overline{x}_0 \in \mathcal{X}$. ■

By Lemma 8, we do not have to consider the general policies π' in the formulation of the stochastic reachability of a target tube problem (4.2).

4.5.3 Convexity: Convex Assumption

With existence conditions established for Assumptions 4 and 5, we now focus on establishing sufficient conditions under which $\mathcal{L}_0^{\pi^*}(\alpha, \mathscr{T})$ is convex.

Assumption 6 (Convex)**.**

a. System dynamics are linear (2.19) and $\mathcal{X} = \mathbb{R}^n$,

b. $\mathcal{U}$ *is convex and compact,*

c. Either

1) Q_k *is input-continuous* $\forall k \in \mathbb{N}_{[0,N-1]}$*, OR*

2) $\mathcal{T}_k$ *is closed* $\forall k \in \mathbb{N}_{[0,N]}$*,*

d. $\mathcal{T}_k$ *is convex* $\forall k \in \mathbb{N}_{[0,N]}$*, and*

e. $\psi_{\boldsymbol{w},k}$ *is a log-concave PDF.*

Since affine transformations of log-concave functions are log-concave, Assumption 6e may also be replaced with the requirement that Q_k is log-concave by (2.22). Under Assumption 6a, 6b, and 6c, the optimization problems in (4.3b) are well-defined and an optimal Markov policy exists (see Remark 4). We will use Proposition 12 in the proof of Theorem 6 to guarantee that the objective of (4.3b) is log-concave (similar to the role played by Proposition 10 in the proof of Theorem 5).

Proposition 12. *Suppose Assumption 6a and 6e holds and* $\mathcal{U}$ *is convex. For every log-concave, Borel-measurable, and non-negative function* $h : \mathcal{X} \to \mathbb{R}$*,* $\int_{\mathcal{X}} h(\overline{y}) Q_k(d\overline{y}|\overline{x},\overline{u})$*,* $\forall k \in \mathbb{N}_{[0,N-1]}$*, is log-concave over* $\mathcal{X} \times \mathcal{U}$*.*

Proof: Similarly to Proposition 10, we show the log-concavity of $\int_{\mathcal{X}} h(\overline{y}) Q_k(d\overline{y}|\overline{x},\overline{u}) = \int_{\mathcal{X}} h(A_k\overline{x} + B_k\overline{u} + \overline{w})\psi_{\boldsymbol{w},k}(\overline{w})d\overline{w}$ $\forall k \in \mathbb{N}_{[0,N-1]}$ over $\mathcal{X} \times \mathcal{U}$. Note that compositions of log-concave functions with affine functions preserve log-concavity [BV04, Sec. 3.2.2]. Hence, $h(A_k\overline{x}_k + B\overline{u}_k + \overline{w}_k)$ is log-concave over $\mathcal{X}\times\mathcal{U}\times\mathcal{W}$. Since multiplication and partial integration preserves log-concavity [BV04, Sec. 3.5.2], we conclude that $\int_{\mathcal{X}} h(\overline{y}) Q_k(d\overline{y}|\overline{x},\overline{u})$, $\forall k \in \mathbb{N}_{[0,N-1]}$ is log-concave over $\mathcal{X} \times \mathcal{U}$. ■

Theorem 6. *Under Assumption 6,*

a. $V_k^*(\cdot)$*,* $\forall k \in \mathbb{N}_{[0,N]}$ *is log-concave over* $\mathcal{X}$*, and*

Property for $k \in \mathbb{N}_{[0,N]}$		f_k (2.17) $\forall k \in \mathbb{N}_{[0,N-1]}$	$\mathcal{X}$ (Borel)	$\mathcal{U}$ (Compact)	$\mathcal{T}_k,\ \forall k \in \mathbb{N}_{[0,N]}$ (Borel)	$Q_k\ \forall k \in \mathbb{N}_{[0,N-1]}$ (Borel-measurable)	Result
$V_k^*(\cdot)$ over $\mathcal{X}$	$\mathcal{L}_k^{\pi^*}(\alpha, \mathscr{T})$						
Measurability		Measurable				Input-continuous	Thm. 4
Piecewise continuity						Continuous	Prop. 9
Upper semi-continuity	Closed $\forall \alpha \in [0,1]$	Continuous	Closed		Closed		Thm. 5
	Compact $\forall \alpha \in (0,1]$				Compact		Prop. 11
Log-concavity	Convex $\forall \alpha \in [0,1]$	Linear (2.19)	Convex	Convex	Convex	Input-continuous & Log-concave	Thm. 6
					Convex & closed	Log-concave	

Table 4.1: Sufficient conditions for various properties of the maximal reach probability $V_k^*(\cdot)$ and the stochastic reach set $\mathcal{L}^{\pi^*}(\cdot)$. See [Aba+07, Thm. 2] for Lipschitz continuity of $V_k^*(\cdot)$.

b. $\mathcal{L}_k^{\pi^*}(\alpha, \mathscr{T}),\ \forall k \in \mathbb{N}_{[0,N]}, \forall \alpha \in [0,1]$ *is convex.*

Proof: The proof of the log-concavity of $V_k^*(\cdot)$ is similar to Theorem 5. The convexity of $\mathcal{T}_k,\ \forall k \in \mathbb{N}_{[0,N]}$ ensures that their respective indicator functions are log-concave. The log-concavity of $V_k^*(\cdot),\ \forall k \in \mathbb{N}_{[0,N]}$ follows from Proposition 12, the fact that log-concavity is preserved under partial supremum over convex sets and multiplication [BV04, Secs. 3.2.5 and 3.5.2], and the convexity of $\mathcal{U}$.

Log-concavity of $V_k^*(\cdot),\ \forall k \in \mathbb{N}_{[0,N]}$ (via quasiconcavity) implies that $\mathcal{L}_k^{\pi^*}(\alpha, \mathscr{T})$, $\forall k \in \mathbb{N}_{[0,N]}$ is convex for $\alpha \in [0,1]$ [BV04, Sec. 3.5]. ■

Remark 5. *With Theorem 6, we have also shown that the dynamic programming solution (4.3) to the stochastic reachability problem of a target tube (4.2) under Assumption 6 is a series of convex optimization problems.*

4.5.4 Polytopic Representation: Convex and Compact Assumption

Theorem 6 and Proposition 11 together guarantee convex and compact $\mathcal{L}_k^{\pi^*}(\alpha, \mathscr{T})$. For ease of discussion, we formulate Assumption 7 to combine the requirements of Assumptions 5 and 6 and Proposition 11.

Assumption 7 (Convex and compact)**.**

a. System dynamics are linear (2.19) *and* $\mathcal{X} = \mathbb{R}^n$,

b. $\mathcal{U}$ *is convex and compact,*

c. $\mathcal{T}_k$ *is convex and compact* $\forall k \in \mathbb{N}_{[0,N]}$, *and*

d. $\psi_{\boldsymbol{w},k}$ *is a log-concave PDF.*

Theorem 7. *Under Assumption 7,* $\mathcal{L}_k^{\pi^*}(\alpha, \mathscr{T}),\ \forall k \in \mathbb{N}_{[0,N]}, \forall \alpha \in (0,1]$ *is convex and compact.*

Proof: Follows from Proposition 11 and Theorem 6. ■

Remark 6. *Assumption 7 enables tight polytopic representation of* $\mathcal{L}_k^{\pi^*}(\alpha, \mathscr{T}),\ \forall k \in \mathbb{N}_{[0,N]}, \alpha \in (0,1]$.

By Proposition 11, if for every $\alpha \in (0,1]$, we require only $\mathcal{L}_0^{\pi^*}(\alpha, \mathscr{T})$ be convex and compact, then Assumption 7c may be relaxed to the following requirements: 1) $\mathcal{T}_0$ is convex and compact, *and* 2) $\mathcal{T}_k,\ \forall k \in \mathbb{N}_{[1,N]}$ is convex and closed.

4.5.5 Underapproximative Interpolation

Next, we address Problem 4 using Theorem 8 under Assumption 7. Theorem 8 states that given the polytopic representations of $\mathcal{L}_k^{\pi^*}(\alpha_1, \mathscr{T})$ and $\mathcal{L}_k^{\pi^*}(\alpha_2, \mathscr{T})$, we can compute the convex combination of the vertices of these polytopes using a specific weight θ to construct a polytopic underapproximation of $\mathcal{L}_k^{\pi^*}(\beta, \mathscr{T})$, $\beta \in [\alpha_1, \alpha_2]$. For a collection of K points $\overline{y}^{(i)} \in \mathbb{R}^n$, recall that $\mathrm{conv}_{i \in \mathbb{N}_{[1,K]}}(\overline{y}^{(i)})$ denotes their convex hull (2.5), which is a polytope.

Theorem 8. *Suppose Assumption 7 holds and let* $k \in \mathbb{N}_{[0,N]}$ *and* $K \in \mathbb{N}, K > 0$. *Given* $\alpha_1, \alpha_2 \in (0,1]$, *let* $\alpha_1 < \alpha_2$, $\overline{x}_1^{(1)}, \ldots, \overline{x}_1^{(K)} \in \mathcal{L}_k^{\pi^*}(\alpha_1, \mathscr{T})$ *and* $\overline{x}_2^{(1)}, \ldots, \overline{x}_2^{(K)} \in \mathcal{L}_k^{\pi^*}(\alpha_2, \mathscr{T})$.

For any $\beta \in [\alpha_1, \alpha_2]$, $\mathrm{conv}_{i \in \mathbb{N}_{[1,K]}}(\overline{y}^{(i)}) \subseteq \mathcal{L}_k^{\pi^*}(\beta, \mathscr{T})$ *where*

$$\overline{y}^{(i)} = \theta \overline{x}_1^{(i)} + (1-\theta)\overline{x}_2^{(i)}, \ \forall i \in \mathbb{N}_{[1,K]}, \ \textit{and} \tag{4.11}$$

$$\theta = \frac{\log(\alpha_2) - \log(\beta)}{\log(\alpha_2) - \log(\alpha_1)} \in [0,1]. \tag{4.12}$$

Proof: By definition of $\overline{x}_1^{(i)}, \overline{x}_2^{(i)}$, $V_k^*\left(\overline{x}_1^{(i)}\right) \geq \alpha_1 > 0$ and $V_k^*\left(\overline{x}_2^{(i)}\right) \geq \alpha_2 > 0$ for every $i \in \mathbb{N}_{[1,K]}$. Note that for θ defined by (4.12), $\theta \in [0,1]$ and $\beta = \alpha_1^{\theta}\alpha_2^{(1-\theta)}$.

Since x^{θ} for $x > 0$ and $\theta \in [0,1]$ is nondecreasing, we have $\left(V_k^*\left(\overline{x}_1^{(i)}\right)\right)^{\theta} \geq \alpha_1^{\theta} > 0$, $\left(V_k^*\left(\overline{x}_2^{(i)}\right)\right)^{(1-\theta)} \geq \alpha_2^{(1-\theta)} > 0$, and $\left(V_k^*\left(\overline{x}_1^{(i)}\right)\right)^{\theta}\left(V_k^*\left(\overline{x}_2^{(i)}\right)\right)^{(1-\theta)} \geq \alpha_1^{\theta}\alpha_2^{(1-\theta)}$ by [Tao06b, Prop. 5.4.7e]. By log-concavity of $V_k^*(\cdot)$ (Theorem 6) and the definition of $\overline{y}^{(i)}$ in (4.11), we have, for every $i \in \mathbb{N}_{[1,K]}$,

$$V_k^*\left(\overline{y}^{(i)}\right) = V_k^*\left(\theta\overline{x}_1^{(i)} + (1-\theta)\overline{x}_2^{(i)}\right) \geq \left(V_k^*\left(\overline{x}_1^{(i)}\right)\right)^{\theta}\left(V_k^*\left(\overline{x}_2^{(i)}\right)\right)^{(1-\theta)} \geq \alpha_1^{\theta}\alpha_2^{(1-\theta)} = \beta.$$

Hence, $\overline{x}^{(1)}, \ldots, \overline{x}^{(K)} \in \mathcal{L}_k^{\pi^*}(\beta, \mathscr{T})$. The proof is completed by the noting that the convex hull of a finite collection of points in a convex set is contained in the set [BV04, Sec. 2.1.4]. ■

We summarize the sufficient conditions for existence, measurability, continuity, and log-concavity of $V_k^*(\cdot)$ and closedness, compactness, and convexity of $\mathcal{L}_k^{\pi^*}(\alpha, \mathscr{T})$ in Table 4.1. Table 4.1 and Theorem 8 addresses Problem 3.

4.6 Underapproximative Verification with Open-Loop Controllers

In stochastic reachability problems, we are typically interested in either an exact computation or an underapproximation. In safety problems, we do not want to overestimate our probability of safety, while underestimating the probability of safety is potentially useful. This trend holds for the stochastic reach set computation as well.

In this section, we use open-loop controllers to compute an underapproximation to maximal reach probability (4.2) and the stochastic reach set (4.5), discuss its compactness and convexity properties, and propose a scalable, grid-free, and anytime algorithm to compute the stochastic reach set.

4.6.1 Open-Loop Controller Synthesis

In [LOE13; VO17], the authors proposed a tractable solution to the stochastic reach-avoid problem by restricting the search for the optimal control policy to open-loop control policies. An open-loop policy $\rho : \mathcal{X} \to \mathcal{U}^N$ provides a sequence of inputs $\rho(\overline{x}_0) = [\overline{u}_0^\top \ \overline{u}_1^\top \ \dots \ \overline{u}_{N-1}^\top]^\top$ for every initial condition $\overline{x}_0$. Note that all actions of this policy are contingent only on the initial state, and not the current state, as in a Markov policy (see Section 2.5.2). The random vector describing the extended state $\boldsymbol{X}$, under the action of $\overline{U} = \rho(\overline{x}_0)$, lies in the probability space $(\mathcal{X}^N, \mathscr{B}(\mathcal{X}^N), \mathbb{P}_{\boldsymbol{X}}^{\overline{x}_0,\overline{U}})$, with $\mathbb{P}_{\boldsymbol{X}}^{\overline{x}_0,\overline{U}}$ defined using Q_k [BS78, Prop. 7.45]. For an initial state $\overline{x}_0 \in \mathcal{T}_0$ (otherwise, the reach probability is zero), the reach probability under $\rho(\cdot)$ is given by

$$r_{\overline{x}_0}^{\rho}(\mathscr{T}) \triangleq \mathbb{P}_{\boldsymbol{X}}^{\overline{x}_0,\overline{U}} \left\{ \forall k \in \mathbb{N}_{[0,N]}, \ \boldsymbol{x}_k \in \mathcal{T}_k \right\}. \tag{4.13}$$

The probability measure $\mathbb{P}_{\boldsymbol{X}}^{\overline{x}_0,\overline{U}}$ in (4.13) is linked to the *forward stochastic reach probability measure* [VHO17; VO18a]. For linear systems, $\mathbb{P}_{\boldsymbol{X}}^{\overline{x}_0,\overline{U}}$ can be computed for arbitrary disturbances using Fourier transforms. Denoting the optimal open-loop controller by ρ^*, we define $W_0^*(\cdot) : \mathcal{X} \to [0,1]$ as,

$$W_0^*(\overline{x}_0) \triangleq r_{\overline{x}_0}^{\rho^*}(\mathscr{T}) = \sup_{\rho(\overline{x}_0)=\overline{U} \in \mathcal{U}^N} r_{\overline{x}_0}^{\rho}(\mathscr{T}). \tag{4.14}$$

where $W_0^* : \mathcal{X} \to [0,1]$ is the maximal reach probability attained by evolving (2.17) from $\overline{x}_0$, when restricted to open-loop controllers. Similarly to (4.3), we define W_k :

$\mathcal{X} \times \mathcal{U}^{N-k} \to [0,1],\ \forall k \in \mathbb{N}_{[0,N-1]}$,

$$W_{N-1}(\overline{x}, \overline{u}_{N-1}) = 1_{\mathcal{T}_{N-1}}(\overline{x}) \int_{\mathcal{X}} 1_{\mathcal{T}_N}(\overline{y}) Q_k(\overline{y}|\overline{x}, \overline{u}_{N-1}) \tag{4.15a}$$

$$W_k(\overline{x}, \overline{U}_{k:N}) = 1_{\mathcal{T}_k}(\overline{x}) \int_{\mathcal{X}} W_{k+1}(\overline{y}, \overline{U}_{k+1:N}) Q_k(\overline{y}|\overline{x}, \overline{u}_k) \tag{4.15b}$$

$$W_0^*(\overline{x}_0) = \sup_{\overline{U} \in \mathcal{U}^N} W_0(\overline{x}_0, \overline{U}) \tag{4.15c}$$

where $\overline{U}_{k:N} = [\overline{u}_k^\top\ \overline{u}_{k+1}^\top\ \dots\ \overline{u}_{N-1}^\top]^\top \in \mathcal{U}^{N-k},\ \forall k \in \mathbb{N}_{[0,N-1]}$, $\overline{U} = \overline{U}_{0:N} \in \mathcal{U}^N$, and $\overline{U}_{N-1:N} = \overline{u}_{N-1} \in \mathcal{U}$. In contrast to $V_k^*(\cdot)$ in (4.3), $W_k(\cdot)$ are not *optimal* value functions, as there is no optimization. Since $r_{\overline{x}_0}^{\rho}(\mathscr{T}) = W_0(\overline{x}_0, \overline{U})$ for $\rho(\overline{x}_0) = \overline{U} \in \mathcal{U}^N$, the optimization problems (4.14) and (4.15c) are equivalent. Similarly to (4.6), we define the α-superlevel set of $r_{\overline{x}_0}^{\rho^*}(\mathscr{T})$ as $\mathcal{K}_0^{\rho^*}(\alpha, \mathscr{T})$,

$$\mathcal{K}_0^{\rho^*}(\alpha, \mathscr{T}) = \{\overline{x}_0 \in \mathcal{X} : r_{\overline{x}_0}^{\rho^*}(\mathscr{T}) \geq \alpha\}. \tag{4.16}$$

Theorem 9 addresses Problem 5.a by carrying forward all the results in Section 4.5 for the open-loop controller-based underapproximations (4.14) and (4.16).

Theorem 9. *a. Under Assumption 4 or Assumption 5, (4.14) is well-posed. Under Assumption 4, $W_0^*(\cdot)$ is Borel-measurable, and under Assumption 5, $W_0^*(\cdot)$ is upper semi-continuous.*

b. Under Assumption 6, (4.14) is a log-concave optimization problem, $W_0(\cdot,\cdot)$ is log-concave over $\mathcal{X} \times \mathcal{U}^N$, and $W_0^*(\cdot)$ is log-concave over $\mathcal{X}$.

c. Under Assumption 7, $\mathcal{K}_0^{\rho^*}(\alpha, \mathscr{T}),\ \forall \alpha \in (0,1]$ is convex and compact.

d. Under Assumption 7, $\mathcal{K}_0^{\rho^*}(\alpha, \mathscr{T}),\ \forall \alpha \in (0,1]$ can be underapproximated by interpolating the vertices of polytopic (underapproximative) representations of $\mathcal{K}_0^{\rho^*}(\alpha_1, \mathscr{T})$ and $\mathcal{K}_0^{\rho^*}(\alpha_2, \mathscr{T})$, as described in Theorem 8.

Proof: Proof of a) with Assumption 4 : Similarly to the proof of Theorem 4, we can show by induction and Definition 2a that $W_0(\overline{x}_0, \overline{U})$ is continuous (and thereby upper semi-continuous) in $\mathcal{U}^N$ for every $\overline{x}_0 \in \mathcal{X}$ when Q_k is input-continuous. Hence, by [HPV76, Thm. 2], we know that (4.15c) (and thereby (4.14)) is well-posed, and an optimal Borel-measurable open-loop controller ρ^* exists and $W_0^*(\cdot)$ is Borel-measurable.

Proof of a) with Assumption 5 : Similarly to the proof of Theorem 5, we can show by induction and Proposition 10 that $W_0(\overline{x}_0, \overline{U})$ is upper semi-continuous in $\mathcal{X} \times \mathcal{U}^N$ and is bounded and nonnegative, when $\mathcal{T}_k$ is closed and f_k is continuous. Hence, by [BS78, Prop. 7.33], we know that (4.15c) (and thereby (4.14)) is well-posed, and an optimal Borel-measurable open-loop controller ρ^* exists, and $W_0^*(\cdot)$ is upper semi-continuous.

Proof of b): Similarly to the proof of Theorem 6, we can show by induction and Proposition 12 that $W_0(\overline{x}_0, \overline{U})$ is log-concave in $\mathcal{X} \times \mathcal{U}^N$ when $\mathcal{T}_k$ is convex and $\psi_{\boldsymbol{w},k}$ is log-concave. Note that $\mathcal{U}^N$ is convex since $\mathcal{U}$ is convex [BV04, Sec. 2.3.2]. Hence, (4.15c) (and thereby (4.14)) is a log-concave optimization. Since partial supremum over convex sets preserves log-concavity [BV04, Sec. 3.2.5, 3.5], $W_0^*(\cdot)$ is log-concave over $\mathcal{X}$.

Proof of c): From Proposition 9a and 9b and the fact that Assumption 7 is a special case of Assumptions 5 and 6 (see Figure 4.2), $\mathcal{K}_0^{\rho^*}(\alpha, \mathscr{T}),\ \forall \alpha \in [0,1]$ is convex and closed by (4.16). Similar to Lemma 6, we note that $\mathcal{K}_0^{\rho^*}(\alpha, \mathscr{T}) \subseteq \mathcal{T}_0$. The compactness assumption of $\mathcal{T}_0$ in Assumption 7 completes the proof, as in Proposition 11.

Proof of d): From Proposition 9c and the discussion in Section 4.5.4, we know that polytopic underapproximations exist for $\mathcal{K}_0^{\rho^*}(\alpha, \mathscr{T})$. The proof, similar to Theorem 8, follows from the log-concavity of W_0^*. ■

4.6.2 Underapproximation Guarantees

We next address Problem 5 using Theorem 10. We first show that the value functions $W_k(\cdot)$ are underapproximations of the optimal value functions $V_k^*(\cdot)$ in Proposition 13.

Proposition 13. *Under Assumption 4 or 5,* $W_k(\overline{x}, \overline{U}_{k:N}) \leq V_k^*(\overline{x})$, $\forall k \in \mathbb{N}_{[0,N-1]}$, $\forall \overline{U}_{k:N} \in \mathcal{U}^{N-k}$ *and* $\forall \overline{x} \in \mathcal{X}$.

Proof: (By induction) We first prove the base case $k = N-1$, i.e., $W_{N-1}(\overline{x}, \overline{U}_{N-1:N}) \leq V_{N-1}^*(\overline{x})$. From (4.3a) and (4.15a), $W_N(\overline{x}) = V_N^*(\overline{x})$, $\forall \overline{x} \in \mathcal{X}$. By [Tao06b, Prop. 19.3.3d], for every $(\overline{x}, \overline{u}) \in \mathcal{X} \times \mathcal{U}$, we have $\int_{\mathcal{X}} W_N(\overline{y}) Q_{N-1}(d\overline{y}|\overline{x}, \overline{u})$ $= \int_{\mathcal{X}} V_N^*(\overline{y}) Q_{N-1}(d\overline{y}|\overline{x}, \overline{u}_{N-1}) \leq \sup_{\overline{u}_{N-1} \in \mathcal{U}} \int_{\mathcal{X}} V_N^*(\overline{y}) Q_{N-1}(d\overline{y}|\overline{x}, \overline{u}_{N-1})$. By (4.3b), (4.15b), and the fact that indicator functions are non-negative, we have $W_{N-1}(\overline{x}, \overline{U}_{N-1:N}) \leq V_{N-1}^*(\overline{x})$ for every $\overline{x} \in \mathcal{X}, \overline{U}_{N-1:N} \in \mathcal{U}$.

Assume for induction, the case $k = t$ ($t \in \mathbb{N}_{[0,N-2]}$) is true, i.e., $W_{t+1}(\overline{x}, \overline{U}_{t+1:N}) \leq V_{t+1}^*(\overline{x})$. By [Tao06b, Prop. 19.3.3c], for every $(\overline{x}, \overline{U}_{t:N}) \in \mathcal{X} \times \mathcal{U}^{N-t}$, we have $\int_{\mathcal{X}} W_{t+1}(\overline{y}, \overline{U}_{t+1:N}) Q_t(d\overline{y}|\overline{x}, \overline{u}_t) = \int_{\mathcal{X}} V_{t+1}^*(\overline{y}) Q_t(d\overline{y}|\overline{x}, \overline{u}_t) \leq \sup_{\overline{u}_t \in \mathcal{U}} \int_{\mathcal{X}} V_N^*(\overline{y}) Q_t(d\overline{y}|\overline{x}, \overline{u}_t)$. The proof is completed by (4.3b), (4.15b), and the fact that indicator functions are non-negative.

We require the assumptions of Assumption 4 or 5 to ensure that (4.14) is well-posed (Theorem 9a). ■

Theorem 10. *Under Assumption 4 or 5,* $W_0^*(\overline{x}) \leq V_0^*(\overline{x})$, $\forall \overline{x} \in \mathcal{X}$, *and* $\mathcal{K}_0^{\rho^*}(\alpha, \mathscr{T}) \subseteq \mathcal{L}_0^{\pi^*}(\alpha, \mathscr{T})$, $\forall \alpha \in [0,1]$.

Proof: By Proposition 13, we know that $W_0(\overline{x}, \overline{U}) \leq V_0^*(\overline{x})$ for every $\overline{x} \in \mathcal{X}$ and $\overline{U} \in \mathcal{U}^N$. By (4.15c) and the definition of the supremum, we have $W_0^*(\overline{x}) = \sup_{\overline{U} \in \mathcal{U}^N} W_0(\overline{x}, \overline{U}) \leq V_0^*(\overline{x})$ Consequently, we have $\mathcal{K}_0^{\rho^*}(\alpha, \mathscr{T}) \subseteq \mathcal{L}_0^{\pi^*}(\alpha, \mathscr{T})$, $\forall \alpha \in [0,1]$ by (4.6) and (4.16). ■

4.7 Underapproximative Verification with Affine Feedback Controllers

4.7.1 Affine Feedback Controller Synthesis

The optimization problem (4.17) constructs an ADF controller (2.23a) to maximize the reach probability. Instead of saturating the ADF controller to satisfy hard control bounds, we require the ADF controller to meet a chance constraint relaxation of the hard control bounds for tractability.

$$\underset{\overline{M},\overline{D}}{\text{minimize}} \quad \mathbb{P}_{\boldsymbol{X}}^{\overline{x}_0,\overline{M},\overline{D}}\{\boldsymbol{X} \in \mathscr{T}\} \tag{4.17a}$$

$$\text{subject to} \quad \boldsymbol{X} = \bar{A}\overline{x}_0 + H\boldsymbol{U} + E\boldsymbol{W} \tag{4.17b}$$

$$\boldsymbol{U} = \overline{M}\boldsymbol{W} + \overline{D}, \quad \overline{M}, \overline{D} \text{ satisfies (2.23)} \tag{4.17c}$$

$$\boldsymbol{W} \sim \mathcal{N}(\overline{\mu}_{\boldsymbol{W}}, C_{\boldsymbol{W}}) \tag{4.17d}$$

$$\mathbb{P}_{\boldsymbol{U}}^{\overline{M},\overline{D}}\left\{\boldsymbol{U} \in \mathcal{U}^N\right\} \geq 1 - \Delta_U \tag{4.17e}$$

where $\Delta_U \in [0, 1)$ is a user-specified threshold for the probabilistic relaxation of the hard control bounds $\boldsymbol{U} \in \mathcal{U}^N$ and initial state $\overline{x}_0 \in \mathcal{T}_0$ (otherwise, the reach probability is zero). The constraint (4.17e) prescribes a lower bound of $1 - \Delta_U$ on the probability that the ADF controller satisfies the hard control bounds. Equivalently, Δ_U is the maximum likelihood with which the ADF controller can violate the hard control bounds. Thus, (4.17) addresses Problem 6.a. Note that setting $\Delta_U = 1$ will trivialize the constraint (4.17e).

We denote the optimal solution of (4.17) as $\overline{M}^*$ and $\overline{D}^*$. The corresponding optimal value function $r_{\overline{x}_0}^{\overline{M}^*,\overline{D}^*}(\mathscr{T}; \Delta_U) : \mathcal{X} \to [0, 1]$ is parameterized by Δ_U,

$$r_{\overline{x}_0}^{\overline{M}^*,\overline{D}^*}(\mathscr{T}; \Delta_U) = \mathbb{P}_{\boldsymbol{X}}^{\overline{x}_0,\boldsymbol{U}}\left\{\boldsymbol{X} \in \mathscr{T} \middle| \boldsymbol{U} = \overline{M}^*\boldsymbol{W} + \overline{D}^*\right\} \tag{4.18}$$

with $\mathbb{P}_{\boldsymbol{U}}^{\overline{M}^*,\overline{D}^*}\left\{\overline{M}^*\boldsymbol{W}+\overline{D}^*\in\mathcal{U}^N\right\}\geq 1-\Delta_U$ due to (4.17e). By construction, $r_{\overline{x}_0}^{\overline{M}^*,\overline{D}^*}(\mathscr{T};\Delta_U)$ is the optimal reach probability obtained using the ADF controller $(\overline{M}^*,\overline{D}^*)$. Note that $r_{\overline{x}_0}^{\overline{M}^*,\overline{D}^*}(\mathscr{T};\Delta_U)$ does not discount the improvement in the reach probability when the ADF controller violates the hard control bounds. Theorem 11 addresses Problem 6.

Theorem 11. *Let Assumption 5 hold. For* $r_{\overline{x}_0}^{\overline{M}^*,\overline{D}^*}(\mathscr{T};\Delta_U)\geq\Delta_U$,

$$r_{\overline{x}_0}^{\pi^*}(\mathscr{T})\geq\frac{r_{\overline{x}_0}^{\overline{M}^*,\overline{D}^*}(\mathscr{T};\Delta_U)-\Delta_U}{1-\Delta_U}. \tag{4.19}$$

Proof: We will first show that

$$r_{\overline{x}_0}^{\pi}(\mathscr{T})\geq\mathbb{P}_{\boldsymbol{X}}^{\overline{x}_0,\overline{M}^*,\overline{D}^*}\left\{\boldsymbol{X}\in\mathscr{T}\middle|\boldsymbol{U}\in\mathcal{U}^N\right\}, \tag{4.20}$$

and then subsequently obtain (4.19) using $r_{\overline{x}_0}^{\overline{M}^*,\overline{D}^*}(\mathscr{T};\Delta_U)$ and Δ_U.

By Lemma 8, there always exist a general policy $\pi'=\boldsymbol{U}$ such that $r_{\overline{x}_0}^{\pi}(\mathscr{T})=r_{\overline{x}_0}^{\pi'}(\mathscr{T})=\mathbb{P}_{\boldsymbol{X}}^{\overline{x}_0,\boldsymbol{U}}\left\{\boldsymbol{X}\in\mathscr{T}\middle|\boldsymbol{U}\in\mathcal{U}^N\right\}$. We have (4.20), since $(\overline{M}^*,\overline{D}^*)$ imposes additional constraints (2.23a) on π' and $\boldsymbol{U}$. Here, $\mathbb{P}_{\boldsymbol{X}}^{\overline{x}_0,\overline{M}^*,\overline{D}^*}\left\{\boldsymbol{X}\in\mathscr{T}\middle|\boldsymbol{U}\in\mathcal{U}^N\right\}$ is the optimal safety probability via the ADF controller $(\overline{M}^*,\overline{D}^*)$ when it simultaneously satisfies the hard control bounds.

By the law of total probability,

$$\begin{aligned}\mathbb{P}_{\boldsymbol{X}}^{\overline{x}_0,\overline{M}^*,\overline{D}^*}\left\{\boldsymbol{X}\in\mathscr{T}\right\}=&\,\mathbb{P}_{\boldsymbol{X}}^{\overline{x}_0,\overline{M}^*,\overline{D}^*}\left\{\boldsymbol{X}\in\mathscr{T}\middle|\boldsymbol{U}\in\mathcal{U}^N\right\}\mathbb{P}_{\boldsymbol{U}}^{\overline{M}^*,\overline{D}^*}\left\{\boldsymbol{U}\in\mathcal{U}^N\right\}\\&+\mathbb{P}_{\boldsymbol{X}}^{\overline{x}_0,\overline{M}^*,\overline{D}^*}\left\{\boldsymbol{X}\in\mathscr{T}\middle|\boldsymbol{U}\notin\mathcal{U}^N\right\}\mathbb{P}_{\boldsymbol{U}}^{\overline{M}^*,\overline{D}^*}\left\{\boldsymbol{U}\notin\mathcal{U}^N\right\}.\end{aligned} \tag{4.21}$$

Since $(\overline{M}^*,\overline{D}^*)$ satisfies (4.17e), $\mathbb{P}_{\boldsymbol{U}}^{\overline{M}^*,\overline{D}^*}\left\{\boldsymbol{U}\in\mathcal{U}^N\right\}=1-\Delta_U+\epsilon_U$ where $\epsilon_U\in[0,\Delta_U]$, accounts for the potential slack in (4.17e). In other words,

$$\mathbb{P}_{\boldsymbol{U}}^{\overline{M}^*,\overline{D}^*}\{\boldsymbol{U}\in\mathcal{U}^N\}=1-(\Delta_U-\epsilon_U),\text{ and }\mathbb{P}_{\boldsymbol{U}}^{\overline{M}^*,\overline{D}^*}\{\boldsymbol{U}\notin\mathcal{U}^N\}=(\Delta_U-\epsilon_U) \tag{4.22}$$

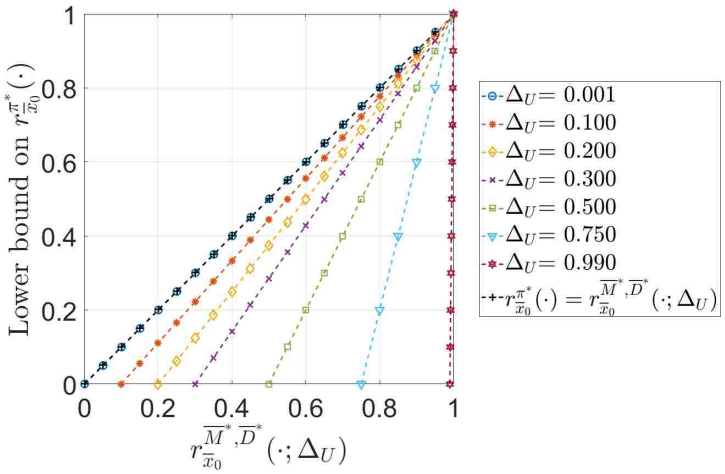


Figure 4.3: Variation of the lower bound (4.19) on $r^{\pi^*}_{\overline{x}_0}(\mathscr{T})$ with the optimal solution to (4.17) $r^{\overline{M}^*,\overline{D}^*}_{\overline{x}_0}(\mathscr{T};\Delta_U)$ for different values of Δ_U. As expected, the lower bound becomes trivial as Δ_U increases.

Since $\mathbb{P}^{\overline{x}_0,\overline{M}^*,\overline{D}^*}_{\boldsymbol{X}}\left\{\boldsymbol{X}\in\mathscr{T}\middle|\boldsymbol{U}\notin\mathcal{U}^N\right\}\in[0,1]$, we can use (4.18), (4.21), and (4.22) to upper bound $r^{\overline{M}^*,\overline{D}^*}_{\overline{x}_0}(\mathscr{T};\Delta_U)$, the optimal value of (4.17),

$$r^{\overline{M}^*,\overline{D}^*}_{\overline{x}_0}(\mathscr{T};\Delta_U)\leq\mathbb{P}^{\overline{x}_0,\overline{M}^*,\overline{D}^*}_{\boldsymbol{X}}\left\{\boldsymbol{X}\in\mathscr{T}\middle|\boldsymbol{U}\in\mathcal{U}^N\right\}(1-(\Delta_U-\epsilon_U))+(\Delta_U-\epsilon_U). \tag{4.23}$$

On rearranging the terms in (4.23) and using the fact that $\epsilon_U\geq 0$,

$$\begin{aligned}\mathbb{P}^{\overline{x}_0,\overline{M}^*,\overline{D}^*}_{\boldsymbol{X}}\left\{\boldsymbol{X}\in\mathscr{T}\middle|\boldsymbol{U}\in\mathcal{U}^N\right\}&\geq 1-\frac{1-r^{\overline{M}^*,\overline{D}^*}_{\overline{x}_0}(\mathscr{T};\Delta_U)}{1-(\Delta_U-\epsilon_U)}\\&\geq 1-\frac{1-r^{\overline{M}^*,\overline{D}^*}_{\overline{x}_0}(\mathscr{T};\Delta_U)}{1-\Delta_U}\\&=\frac{r^{\overline{M}^*,\overline{D}^*}_{\overline{x}_0}(\mathscr{T};\Delta_U)-\Delta_U}{1-\Delta_U}.\end{aligned} \tag{4.24}$$

The requirement of $r^{\overline{M}^*,\overline{D}^*}_{\overline{x}_0}(\mathscr{T};\Delta_U)\geq\Delta_U$ ensures that the lower bound (4.24) is nonnegative. We have (4.19) from (4.20) and (4.24). ■

Figure 4.3 shows the lower bounds on $r^{\pi^*}_{\overline{x}_0}(\mathscr{T})$ produced by (4.19) for different values of Δ_U and $r^{\overline{M}^*,\overline{D}^*}_{\overline{x}_0}(\mathscr{T};\Delta_U)$. As an example, we obtain a lower bound for $r^{\pi^*}_{\overline{x}_0}(\mathscr{T})$

of 0.75 for $\Delta_U = 0.2$ and $r_{\overline{x}_0}^{\overline{M}^*,\overline{D}^*}(\mathscr{T};\Delta_U) = 0.8$. Intuitively, using smaller values of Δ_U (lower violation probability) should provide more informative lower bounds on $r_{\overline{x}_0}^{\pi^*}(\mathscr{T})$. By (4.17e), a smaller value of Δ_U forces the affine disturbance feedback to be more "compliant" with the hard control bounds. From Figure 4.3, we see an increase in the rate at which the lower bound (4.19) diminishes to zero (trivial lower bound) as Δ_U increases (relaxation of the control bounds).

4.8 Summary

This chapter provides the theoretical framework for the problem of stochastic reachability of a target tube. Specifically, we identify sufficient conditions under which the stochastic reach set is closed, bounded, compact, and convex. Using these convexity and compactness properties, we describe an underapproximative interpolation technique for the stochastic reach sets. We also construct underapproximative guarantees for the maximal reach probability using point-based stochastic reachability evaluations (open-loop and affine feedback controller synthesis).

Chapter 5

Stochastic Reachability of a Target Tube: Computation

5.1 Introduction

This chapter builds on the results presented in Chapter 4 to propose four novel approaches for controller synthesis for the stochastic reachability of a target tube. Our approaches are grid-free and scale well with dimension, when compared to the current state-of-the-art dynamic programming-based approaches [Aba+07; Aba+08; SL10]. Our algorithms rely on convex optimization, stochastic programming, and Fourier transforms to compute underapproximations to the maximal reach-probability as well as synthesize open-loop or affine-feedback controllers. We also combine these point-based verification and controller synthesis techniques with a ray-shooting algorithm (Figure 2.1) to underapproximate the sets in a scalable, grid-free, and anytime algorithm. This approach enables, for the first time, the verification of systems as high as 40-dimensions. All algorithms presented in this chapter has been implemented in `SReachTools`, an open-source MATLAB toolbox (see Chapter 6).

Recall that Gaussian probability density is log-concave [BV04, Sec. 3.5]. We will make the following assumption (a special case of Assumption 7) for tractability.

Assumption 8. *We presume the dynamics are linear (potentially time-varying)*

(2.19) *with (not necessarily independent or identical) Gaussian disturbance* $\boldsymbol{w}_k$*, and the target sets that compose the target tube* $\mathscr{T}$ *and the input space* $\mathcal{U}$ *are polytopic.*

Under this assumption, the concatenated disturbance random vector $\boldsymbol{W} = [\boldsymbol{w}_0^\top \ \boldsymbol{w}_1^\top \ \dots \ \boldsymbol{w}_{N-1}^\top]^\top$ is also a Gaussian random vector [Gub06, Ch. 9]. Here, $\boldsymbol{W} \sim \mathcal{N}(\overline{\mu}_{\boldsymbol{W}}, C_{\boldsymbol{W}})$ with $\overline{\mu}_{\boldsymbol{W}} \in \mathbb{R}^{pN}$ and $C_{\boldsymbol{W}} \in \mathbb{R}^{pN \times pN}$; $C_{\boldsymbol{W}}$ is positive semi-definite, and $\mathbb{P}_{\boldsymbol{W}}$ denotes the probability measure of $\boldsymbol{W}$. Due to the linearity of the system (2.19), $\boldsymbol{X}$ is also Gaussian [Gub06, Sec. 9.2]. The stochasticity of $\boldsymbol{X}$ and $\boldsymbol{U}$ is given (2.27) and (2.28) respectively. Additionally, $\boldsymbol{X} \in \mathscr{T}$ and $\boldsymbol{U} \in \mathcal{U}^N$ are equivalent to $G\boldsymbol{X} \leq \overline{h}$ and $P\boldsymbol{X} \leq \overline{q}$ for some known $G = [\overline{g}_0^\top \ \overline{g}_1^\top \ \cdots \overline{g}_{L_X}^\top]^\top \in \mathbb{R}^{L_X \times nN}$, $\overline{h} \in \mathbb{R}^{L_X}$, $P = [\overline{p}_0^\top \ \overline{p}_1^\top \ \cdots \overline{p}_{L_U}^\top]^\top \in \mathbb{R}^{L_U \times mN}$, $\overline{q} \in \mathbb{R}^{L_U}$, $L_X, L_U \in \mathbb{N}$, and $L_X, L_U > 0$.

5.2 Related Work

We can compute the maximal reach probability and the reach sets for low-dimensional systems using dynamic programming by gridding the state space [SB98; Aba+07]. The reliance on a grid over the state space translates to an exponentially increasing computational cost as the system dimension increases, making this approach intractable for system dimensions higher than three or four. Researchers have focused on alleviating this *curse of dimensionality* via approximate dynamic programming [KML16; Man+15], Gaussian mixtures [KML16], particle filters [Man+15; LOE13], convex chance-constrained optimization [LOE13], Fourier transforms and open-loop controllers [VO17; VO18b], set-theoretic (Lagrangian) approaches [GVO17], and semi-definite programming [Drz+16; Kar+17].

With respect to controller synthesis, almost all of these approaches seek open-loop control laws for tractability, at the cost of significant conservativeness [VO17; LOE13]. Approximate dynamic programming techniques can synthesize closed-loop controllers by parameterizing the policy space [Man+15; Kar+17]. However, these

approximation approaches do not guarantee an underapproximation of the safety probability. This limits its application to safety-critical applications, where we can not afford to be overconfident with the degree of safety.

In the past few decades, the controller synthesis problem for stochastic optimal control problems has received a lot of attention. Specifically, techniques based on stochastic receding horizon control (also known as stochastic model predictive control) have been developed that can synthesize (sub)optimal controllers to drive the system while satisfying the stochastic dynamics, probabilistic state and input constraints, and minimizing a cost function. See [Mes16; FGS16] for recent surveys on this problem. However, these techniques can not be used to compute the set of good initial states which is the main focus of this chapter.

A related problem of robust reachability of a target tube has been discussed in [BR71; Ber72]. Here, the state of a bounded non-stochastic uncertainty-perturbed system must be steered to lie within a target tube, for all possible realizations of the uncertainty. This work utilizes computational geometry to construct the reach sets, and has been extended for the case of stochastic reach-avoid problems in [GVO17]. However, its reliance on vertex-facet enumeration precludes computation on problems with large time horizons or small sets in the target tube. Other techniques using computational geometry (ellipsoids, zonotopes, and support functions) are available for linear systems [Gir05; KV00; LGG09]. For the case of non-stochastic uncertainty, several approaches based on robust optimal control exist to identify the backward reach sets. For continuous time systems, techniques using Hamilton-Jacobi formulation and level set techniques have been proposed [MBT05].

5.3 Problem Statements

Problem 7. *Solve the chance-constraint formulation (5.8) for affine controller synthesis using difference of convex programming and piecewise affine approximations.*

Problem 7.a. *Propose an algorithm to construct a piecewise-affine overapproximation of the convex function $\Phi^{-1}(1-z)$ for $z \in [\delta_{\text{lb}}, 0.5]$ and some $\delta_{\text{lb}} > 0$.*

Problem 7.b. *Cast the chance constraint optimization problem obtained in Problem 6.a by reformulating it into an equivalent difference of convex program.*

Problem 7.a is motivated by the need for a conic representation of (5.8e) and (5.8f). This will enable the use of standard convex solvers. Recall that two optimization problems are equivalent if an optimal solution to one can be used to readily find an optimal solution to the other [BV04, Sec. 4.1.3].

Problem 8. *Discuss three approaches to perform open-loop controller synthesis using:*

1. *Fourier transforms and log-concave optimization (Theorem 9b).*
2. *a chance-constraint formulation and piecewise-affine approximations, where (5.8) is reformulated as a linear program with $\overline{M} = 0$.*
3. *a mixed-integer linear program formulation, with the scenarios undersampled using Voronoi partitions.*

Problem 9. *Propose a scalable, grid-free, and anytime algorithm to compute a open-loop controller-based polytopic underapproximation to $\mathcal{L}_0^{\pi^*}(\alpha, \mathscr{T})$, $\forall \alpha \in [0,1]$.*

5.4 Piecewise-Affine Overapproximation of a Convex Function

Recall that $\Phi^{-1}(1-z)$ is convex $z \in [0, 0.5]$. The convexity of $\Phi^{-1}(1-z)$ follows from its non-negative Hessian over $z \in [0, 0.5]$ [BV04, Sec. 3.1]). An intuition on its

convexity may also be obtained by visual inspection in Figure 5.1. The convexity of $\Phi^{-1}(1-z)$ implies the following constraint with decision variables $z \in [0, 0.5], s \geq 0$ is convex,

$$\Phi^{-1}(1-z) \leq s, \quad z \in [0, 0.5]. \tag{5.1}$$

Note that (5.1) does not have a exact conic reformulation. This prevents us from using standard convex solvers like Gurobi [Gur15], when solving convex optimization problems with constraints of the form (5.1). To address this challenge, we seek to construct $\ell_\Phi^+(z)$, a piecewise affine overapproximation of $\Phi^{-1}(1-z)$. This ensures that (5.1) can be enforced using a collection of linear constraints, since

$$\ell_\Phi^+(z) \leq s \Rightarrow \Phi^{-1}(1-z) \leq s \tag{5.2}$$

for any $z \in [0, 0.5], s \geq 0$. We restrict the domain of the piecewise-affine overapproximation $\ell_\Phi^+(z)$ of $\Phi^{-1}(1-z)$ to $z \in [\delta_{\text{lb}}, 0.5]$ for a small $\delta_{\text{lb}} \in \mathbb{R}, \delta_{\text{lb}} > 0$ since $\Phi^{-1}(1-z) \to \infty$ as $z \to 0^+$. For a user-specified maximum overapproximation error $\eta > 0$, we wish to compute $m_j^+, c_j^+ \in \mathbb{R}$, and $N_\Phi \in \mathbb{N}$ and $N_\Phi > 0$, such that

$$\ell_\Phi^+(z) \triangleq \max_{j \in \mathbb{N}_{[1,N_\Phi]}} \{m_j^+ z + c_j^+\}, \text{ and } 0 \leq \ell_\Phi^+(z) - \Phi^{-1}(1-z) \leq \eta, \ \forall z \in [\delta_{\text{lb}}, 0.5]. \tag{5.3}$$

Given $z_1 \in [\delta_{\text{lb}}, 0.5]$ and $h > 0$, we define $\ell_\Phi^+(z)$ for $z \in [z_1, z_1 + h]$ as the line segment joining $(z_1, \Phi^{-1}(1-z_1))$ and $(z_1 + h, \Phi^{-1}(1-(z_1+h)))$. By the Lagrange remainder theorem [AS65, Eq. 25.2.3], there exists $z_3 \in [z_1, z_1 + h]$ such that for every $z \in [z_1, z_1 + h]$,

$$\ell_\Phi^+(z) - \Phi^{-1}(1-z) \leq \frac{(z-z_1)(z_1+h-z)\nabla^2\Phi^{-1}(1-z_3)}{2}. \tag{5.4}$$

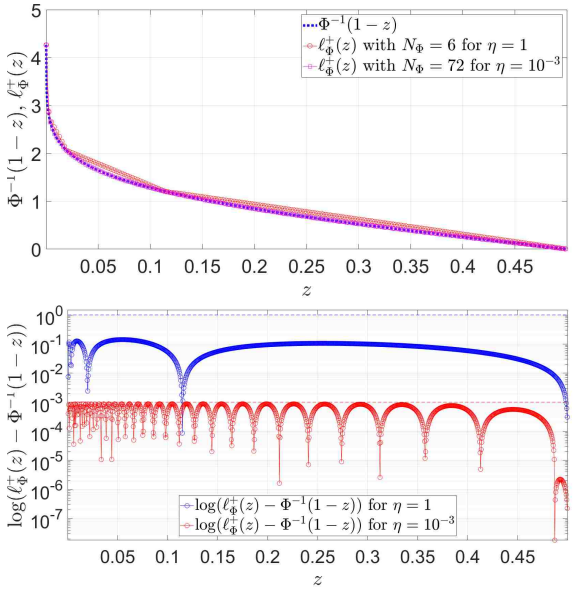


Figure 5.1: Piecewise-affine overapproximation $\ell^+_\Phi(z)$ of the function $\Phi^{-1}(1-z)$ for a maximum overapproximation error $\eta \in \{10^{-3}, 1\}$. The plot above compares the $\ell^+_\Phi(z)$ with $\Phi^{-1}(1-z)$. The plot below shows the overapproximation error.

We simplify (5.4) to a simple upper bound on the overapproximation error by maximizing the right hand size of (5.4) for $z, z_3 \in [z_1, z_1 + h]$. Since the Hessian of $\Phi^{-1}(1-z)$ is monotone decreasing in z, $\max_{z_3 \in [z_1, z_1+h]} \nabla^2 \Phi^{-1}(1-z_3) = \Phi^{-1}(1-z_1)$. In addition, we have $\max_{z \in [z_1, z_1+h]} (z - z_1)(z_1 + h - z) = h^2/4$. Therefore, for every $z \in [z_1, z_1 + h]$,

$$0 \leq \ell^+_\Phi(z) - \Phi^{-1}(1-z) \leq \frac{h^2}{8} \nabla^2 \Phi^{-1}(1-z_1) \tag{5.5}$$

by (5.4) and the convexity of $\Phi^{-1}(1-z)$ [BV04, Sec. 3.1.1].

Algorithm 5 solves for z_1, h iteratively by starting from $z_1 = \delta_{\text{lb}}$ and setting the error bound (5.5) to the user-specified threshold η. Figure 5.1 shows the quality of the piecewise-affine overapproximation of the function $\Phi^{-1}(1-z)$ for $\eta \in \{10^{-3}, 1\}$ obtained via Algorithm 5. Algorithm 5 addresses Problem 7.a.

Note that the ideas discussed in this section can be easily generalized to compute a piecewise-affine overapproximation for an arbitrary convex function. Further, using

Algorithm 5 Construction of a piecewise-affine overapproximation of $\Phi^{-1}(1-z)$

Input: A maximum overapproximation error $\eta > 0$
Output: Piecewise affine overapproximation $\ell_\Phi^+(z)$
1: $z_0 \leftarrow \delta_{\text{lb}}$ and $j \leftarrow 0$
2: **while** $z_j < 0.5$ **do**
3: $\quad h \leftarrow \min\left(\sqrt{\frac{8*\eta}{\nabla^2\Phi^{-1}(1-z_1)}}, z_{\max} - z_j\right)$, $z_{j+1} \leftarrow z_j + h$
4: $\quad m_j^+ \leftarrow \frac{\Phi^{-1}(1-z_{j+1})-\Phi^{-1}(1-z_j)}{h}$, $c_j^+ \leftarrow \frac{\Phi^{-1}(1-z_{j+1})z_j-(z_{j+1})\Phi^{-1}(1-z_j)}{h}$
5: $\quad$ Increment j by 1
6: **end while**
7: $N_\Phi \leftarrow j$, $\ell_\Phi^+(z) \leftarrow \max_{j\in\mathbb{N}_{[1,N_\Phi]}} \{m_j^+ z + c_j^+\}$

first-order Taylor approximation, one can construct a piecewise affine underapproximation of a convex function. Similar approaches can be used for approximation of concave functions. See [VSO19] for more details.

5.5 Affine Feedback Controller Synthesis: Computation

5.5.1 Affine Feedback Controller Synthesis via Chance Constraints-Based Restriction

Motivated by [LOE13; Old+14], we consider (5.6), a well-studied restriction of chance-constraint problems of the form (4.17). This reformulation is known as the quantile function-based reformulation [Old+14] or risk allocation technique [VT11].

$$\text{minimize} \quad \sum_{i=1}^{L_X} \alpha_i \tag{5.6a}$$

$$\text{subject to} \quad \overline{M}, \overline{D} \text{ satisfies (2.23)} \tag{5.6b}$$

$$\overline{\mu}_{\boldsymbol{X}}^{\overline{x}_0,\overline{M},\overline{D}}, \overline{\mu}_{\boldsymbol{U}}^{\overline{M},\overline{D}}, C_{\boldsymbol{X}}^{\overline{M}}, C_{\boldsymbol{U}}^{\overline{M}} \text{ satisfy (2.27b) and (2.28b)} \tag{5.6c}$$

$$\overline{\alpha} \in [0, 0.5]^{L_X}, \ \overline{\beta} \in [0, 0.5]^{L_U}, \ \overline{x}_0 \in \mathcal{T}_0 \tag{5.6d}$$

$$\sum_{i=1}^{L_X} \alpha_i \leq 1 - \Delta_U, \ \sum_{i=1}^{L_U} \beta_i \leq \Delta_U \tag{5.6e}$$

$$\overline{g}_i^\top \overline{\mu}_{\boldsymbol{X}}^{\overline{x}_0,\overline{M},\overline{D}} + \sqrt{\overline{g}_i^\top C_{\boldsymbol{X}}^{\overline{M}} \overline{g}_i}\Phi^{-1}(1-\alpha_i) \leq h_i, \ \forall i \in \mathbb{N}_{[1,L_X]} \tag{5.6f}$$

$$\overline{p}_i^\top \overline{\mu}_{\boldsymbol{U}}^{\overline{M},\overline{D}} + \sqrt{\overline{p}_i^\top C_{\boldsymbol{U}}^{\overline{M}} \overline{p}_i}\Phi^{-1}(1-\beta_i) \leq q_i, \ \forall i \in \mathbb{N}_{[1,L_U]} \tag{5.6g}$$

with decision variables $\overline{M}$, $\overline{D}$, $\overline{\alpha}$, $\overline{\beta}$, $\overline{\mu}_{\boldsymbol{X}}^{\overline{x}_0,\overline{M},\overline{D}}$, $\overline{\mu}_{\boldsymbol{U}}^{\overline{M},\overline{D}}$, $C_{\boldsymbol{X}}^{\overline{M}}$, and $C_{\boldsymbol{U}}^{\overline{M}}$. Here, $\overline{\alpha} = [\alpha_1\ \alpha_2\ \ldots\ \alpha_{L_X}]^\top$ and $\overline{\beta} = [\beta_1\ \beta_2\ \ldots\ \beta_{L_X}]^\top$ are maximum violation probability thresholds for the linear constraints $G\boldsymbol{X} \leq \overline{h}$ and $P\boldsymbol{U} \leq \overline{q}$, and $\Phi^{-1}(\cdot)$ is the inverse of the standard normal cumulative density function, also known as the quantile function.

Proposition 14. *Every feasible solution of* (5.6) *is a feasible solution of* (4.17) *with* $r_{\overline{x}_0}^{\overline{M}^*,\overline{D}^*}(\mathscr{T};\Delta_U) \geq 1 - \sum_{i=1}^{L_X} \alpha_i^*$.

Proposition 14 follows from the fact that: 1) Gaussian random vectors are completely characterized by their mean and covariance, enforced in (5.6c), and 2) we can conservatively approximate constraints of the form (4.17e) using (5.6a) and (5.6d)–(5.6g) via Boole's inequality and properties of the Gaussian random vector. A constraint similar to (4.17e) is used for $\boldsymbol{X}$ instead of (4.17a) by reformulating (4.17) in the epigraph form. The constraint $\sum_{i=1}^{L_X} \alpha_i \leq 1 - \Delta_U$ arises from Theorem 11, which requires $\Delta_U \leq r_{\overline{x}_0}^{\overline{M}^*,\overline{D}^*}(\mathscr{T};\Delta_U)$. For any random vector $\boldsymbol{z}$ with probability measure $\mathbb{P}_{\boldsymbol{z}}$ and Borel sets $\{\mathcal{A}_i\}$ with $\mathcal{A} = \cup_i \mathcal{A}_i$, Boole's inequality states that

$$\mathbb{P}_{\boldsymbol{z}}\{\boldsymbol{z} \in \mathcal{A}\} \leq \sum_i \mathbb{P}_{\boldsymbol{z}}\{\boldsymbol{z} \in \mathcal{A}_i\}. \tag{5.7}$$

5.5.2 Reformulation to a Difference of Convex Program

The constraints (5.6f) and (5.6g) involve products of convex functions of the decision variable, which makes asserting their convexity challenging. Therefore, we formulate

the following equivalent difference of convex program,

$$\text{minimize} \quad \sum_{i=1}^{L_X} \alpha_i \tag{5.8a}$$

$$\text{subject to} \quad (5.6b), (5.6c), (5.6d), (5.6e) \tag{5.8b}$$

$$\sqrt{\overline{g}_i^\top C_{\boldsymbol{X}}^{\overline{M}} \overline{g}_i} \leq s_{\boldsymbol{X},i}, \ \forall i \in \mathbb{N}_{[1,L_X]} \tag{5.8c}$$

$$\sqrt{\overline{p}_i^\top C_{\boldsymbol{U}}^{\overline{M}} \overline{b}_i} \leq s_{\boldsymbol{U},i}, \ \forall i \in \mathbb{N}_{[1,L_U]} \tag{5.8d}$$

$$\Phi^{-1}(1-\alpha_i) \leq t_{\boldsymbol{X},i}, \ \forall i \in \mathbb{N}_{[1,L_X]} \tag{5.8e}$$

$$\Phi^{-1}(1-\beta_i) \leq t_{\boldsymbol{U},i}, \ \forall i \in \mathbb{N}_{[1,L_U]} \tag{5.8f}$$

$$\left(\overline{g}_i^\top \overline{\mu}_{\boldsymbol{X}}^{\overline{x}_0,\overline{M},\overline{D}} + \frac{(s_{\boldsymbol{X},i}+t_{\boldsymbol{X},i})^2}{2}\right) - \left(h_i + \frac{s_{\boldsymbol{X},i}^2 + t_{\boldsymbol{X},i}^2}{2}\right) \leq 0, \ \forall i \in \mathbb{N}_{[1,L_X]} \tag{5.8g}$$

$$\left(\overline{p}_i^\top \overline{\mu}_{\boldsymbol{U}}^{\overline{M},\overline{D}} + \frac{(s_{\boldsymbol{U},i}+t_{\boldsymbol{U},i})^2}{2}\right) - \left(q_i + \frac{s_{\boldsymbol{U},i}^2 + t_{\boldsymbol{U},i}^2}{2}\right) \leq 0, \ \forall i \in \mathbb{N}_{[1,L_U]} \tag{5.8h}$$

with decision variables $\overline{M}$, $\overline{D}$, $\overline{\alpha}$, $\overline{\beta}$, $\overline{\mu}_{\boldsymbol{X}}^{\overline{x}_0,\overline{M},\overline{D}}$, $\overline{\mu}_{\boldsymbol{U}}^{\overline{M},\overline{D}}$, $C_{\boldsymbol{X}}^{\overline{M}}$, $C_{\boldsymbol{U}}^{\overline{M}}$, $\overline{s}_{\boldsymbol{X}}$, $\overline{s}_{\boldsymbol{U}}$, $\overline{t}_{\boldsymbol{X}}$, and $\overline{t}_{\boldsymbol{U}}$. Here, $\overline{s}_{(\cdot)} = [s_{(\cdot),1} \ s_{(\cdot),2} \ \dots \ s_{(\cdot),L_{(\cdot)}}] \in \mathbb{R}^{L_{(\cdot)}}$ and $\overline{t}_{(\cdot)} = [t_{(\cdot),1} \ t_{(\cdot),2} \ \dots \ t_{(\cdot),L_{(\cdot)}}] \in \mathbb{R}^{L_{(\cdot)}}$.

Theorem 12. *Problem (5.8) is a difference of convex program, and it is equivalent to (5.6).*

Proof: *Difference of convex program*: The cost (5.8a) and the constraints (5.8b)–(5.8f) are convex. The constraints (5.8g) and (5.8h) are difference of convex constraints of the form (2.14b), since sums of convex functions and composition of convex functions with affine transformations are convex [BV04, Sec. 3.2].

Equivalence: As discussed in Section 2.4.2, the constraint (5.8g) simplifies to $\overline{g}_i^\top \overline{\mu}_{\boldsymbol{X}}^{\overline{x}_0,\overline{M},\overline{D}} + s_{\boldsymbol{X},i} t_{\boldsymbol{X},i} \leq h_i$. From (5.8c) and (5.8e), the constraint (5.6f) is related to

the constraint (5.8g) by (using $\overline{\mu}_{\boldsymbol{X}}$ for $\overline{\mu}_{\boldsymbol{X}}^{\overline{x}_0,\overline{M},\overline{D}}$)

$$\overline{g}_i^\top \overline{\mu}_{\boldsymbol{X}} + \sqrt{\overline{g}_i^\top C_{\boldsymbol{X}}^{\overline{M}} \overline{g}_i}\,\Phi^{-1}(1-\alpha_i) \leq \overline{g}_i^\top \overline{\mu}_{\boldsymbol{X}} + s_{\boldsymbol{X},i} t_{\boldsymbol{X},i} \leq h_i. \tag{5.9}$$

A similar relation can be derived for the input chance constraints (5.6g) using (5.8d) and (5.8f) as well. All other constraints (5.8b) and the objective functions are identical. Therefore, every optimal solution of (5.6) is an optimal solution of (5.8) with the values for the slack variables $\overline{s}_{(.)}$ and $\overline{t}_{(.)}$ computed from (5.8c)–(5.8f). Similarly, every optimal solution of (5.8) satisfies (5.6) by (5.9). Thus, (5.6) and (5.8) are equivalent optimization problems. ■

Theorem 12 addresses Problem 7.b.

5.5.3 Convexification to a Second-Order Cone Problem

Algorithm 1 iteratively convexifies the non-convex problem (2.14) by linearizing the non-convex terms about a solution iterate. In its current form, we can not solve (5.8) using Algorithm 1 and standard convex solvers. This is because the constraints (5.8e) and (5.8f) do not have a conic reformulation. Using (5.2) and (5.3), we tighten the constraints (5.8e) and (5.8f) and replace it with their piecewise-affine restrictions,

$$m_j^+ \alpha_i + c_j^+ \leq t_{\boldsymbol{X},i}, \quad \forall i \in \mathbb{N}_{[1,L_X]}, \forall j \in \mathbb{N}_{[1,N_\Phi]} \tag{5.10a}$$

$$m_j^+ \beta_i + c_j^+ \leq t_{\boldsymbol{U},i}, \quad \forall i \in \mathbb{N}_{[1,L_U]}, \forall j \in \mathbb{N}_{[1,N_\Phi]}. \tag{5.10b}$$

For sake of clarity, we include the convex subproblem, corresponding to (5.8), that needs to be solved iteratively as done in Algorithm 1.

$$\text{minimize} \quad \sum\nolimits_{i=1}^{L_X} \alpha_i \tag{5.11a}$$

$$\text{subject to} \quad (5.6b), (5.6c), (5.6d), (5.6e), (5.10a), (5.10b) \tag{5.11b}$$

$$\left\| \left(C_{\boldsymbol{X}}^{\overline{M}}\right)^{\frac{1}{2}} \overline{g}_i \right\|_2 \leq s_{\boldsymbol{X},i}, \ \forall i \in \mathbb{N}_{[1,L_X]}, \quad \left\| \left(C_{\boldsymbol{U}}^{\overline{M}}\right)^{\frac{1}{2}} \overline{p}_j \right\|_2 \leq s_{\boldsymbol{U},j}, \ \forall j \in \mathbb{N}_{[1,L_U]} \tag{5.11c}$$

$$\overline{g}_i^\top \overline{\mu}_{\boldsymbol{X}}^{\overline{x}_0,\overline{M},\overline{D}} + \frac{(s_{\boldsymbol{X},i} + t_{\boldsymbol{X},i})^2}{2} - h_i - \frac{\left(s_{\boldsymbol{X},i}^{(k)}\right)^2 + \left(t_{\boldsymbol{X},i}^{(k)}\right)^2}{2} - s_{\boldsymbol{X},i}^{(k)}(s_{\boldsymbol{X},i} - s_{\boldsymbol{X},i}^{(k)}) - t_{\boldsymbol{X},i}^{(k)}(t_{\boldsymbol{X},i} - t_{\boldsymbol{X},i}^{(k)}) \leq 0, \ \forall i \in \mathbb{N}_{[1,L_X]} \tag{5.11d}$$

$$\overline{p}_i^\top \overline{\mu}_{\boldsymbol{U}}^{\overline{M},\overline{D}} + \frac{(s_{\boldsymbol{U},i} + t_{\boldsymbol{U},i})^2}{2} - q_i - \frac{\left(s_{\boldsymbol{U},i}^{(k)}\right)^2 + \left(t_{\boldsymbol{U},i}^{(k)}\right)^2}{2} - s_{\boldsymbol{U},i}^{(k)}(s_{\boldsymbol{U},i} - s_{\boldsymbol{U},i}^{(k)}) - t_{\boldsymbol{U},i}^{(k)}(t_{\boldsymbol{U},i} - t_{\boldsymbol{U},i}^{(k)}) \leq 0, \ \forall i \in \mathbb{N}_{[1,L_U]} \tag{5.11e}$$

with decision variables $\overline{M}$, $\overline{D}$, $\overline{\alpha}$, $\overline{\beta}$, $\overline{\mu}_{\boldsymbol{X}}^{\overline{x}_0,\overline{M},\overline{D}}$, $\overline{\mu}_{\boldsymbol{U}}^{\overline{M},\overline{D}}$, $C_{\boldsymbol{X}}^{\overline{M}}$, $C_{\boldsymbol{U}}^{\overline{M}}$, $\overline{s}_{\boldsymbol{X}}$, $\overline{s}_{\boldsymbol{U}}$, $\overline{t}_{\boldsymbol{X}}$, and $\overline{t}_{\boldsymbol{U}}$. Here, $\overline{s}_{(\cdot)}^{(k)}, \overline{t}_{(\cdot)}^{(k)}$ refers to the slack variable values at different iterations in Algorithm 1. Note that (5.11) is a second-order cone problem. The constraints (5.11d) and (5.11e) can be reformulated as second-order cone constraints using the hyperbolic constraint [BV04, Ex. 4.26]. For a vector $\overline{x} \in \mathbb{R}^n$ and scalars $y, z \in \mathbb{R}$ (all can be decision variables), the constraint

$$\overline{x}^\top \overline{x} \leq yz \iff \left\| \begin{bmatrix} 2\overline{x}^\top & y - z \end{bmatrix}^\top \right\| \leq y + z. \tag{5.12}$$

Second order cone problems can be solved efficiently using standard convex solvers [GB14; Gur15]. Problem (5.11) along with Algorithm 1 addresses Problem 7.

5.5.4 Implementation Details

The user must specify the following parameters: an initial value for the decision variables involved in the DC constraints $\overline{s}_{(\cdot)}^{(0)}$ and $\overline{t}_{(\cdot)}^{(0)}$, the scaling sequence for the slack penalty $(\tau_0, \gamma, \tau_{\max})$, the parameters for the exit condition ϵ_{dc} and ϵ_{viol}, and the user-defined overapproximation for η.

We repeat our recommendations for these parameters, as described in [VO19]. We initialize $\overline{s}_{(\cdot)}^{(0)}$ and $\overline{t}_{(\cdot)}^{(0)}$ by solving the constraints (5.8c)–(5.8f) with zero feedback gain $\overline{M} = \overline{0}_{nN \times pN}$, and equally distributed risk allocations $\overline{\alpha} = \min\left(\frac{1-\Delta_U}{L_X}, 0.5\right)\overline{1}_{L_X \times 1}$ and $\overline{\beta} = \min\left(\frac{\Delta_U}{L_U}, 0.5\right)\overline{1}_{L_U \times 1}$. Here, $\overline{z}_{a \times b}$ is a $a \times b$-dimensional matrix containing the value $z \in \mathbb{R}$. Alternatively, we can initialize these variables randomly [LB16, Sec. 1.2]. For the remaining parameters of Algorithm 1, we chose $\tau_0 = 1$, $\gamma = 2$, $\tau_{\max} = 10^5$, $\epsilon_{\text{dc}} = 10^4$, and $\epsilon_{\text{viol}} = 10^{-8}$. While different values for these parameters result in different convergence rates, there is no clear consensus on how best to choose these parameters. Finally, we chose the maximum overapproximation error for the piecewise-affine overapproximation of $\Phi^{-1}(1-z)$ as $\eta = 10^{-3}$. As seen in Figure 5.1, reducing η decreases the conservativeness introduced by the piecewise-affine approximation. However, it increases N_Φ, which in turn increases the computational effort required to solve (5.8).

5.6 Open-Loop Controller Synthesis

In this section, we address Problem 8 by synthesizing open-loop controllers $\overline{U} \in \mathcal{U}^N$ that maximize safety probability. Recall that affine controllers (2.23a) simplify to open-loop controllers when $\overline{M} = 0$. By (2.26) and (2.25), given an initial state $\overline{x}_0 \in \mathcal{X}$ and an open-loop vector $\overline{U} \in \mathcal{U}^N$, the concatenated state random vector $\boldsymbol{X}$

is a Gaussian under Assumption 8,

$$\boldsymbol{X} \sim \mathcal{N}(\overline{\mu}_{\boldsymbol{X}}^{\overline{x}_0,\overline{U}}, C_{\boldsymbol{X}}), \tag{5.13a}$$

$$\overline{\mu}_{\boldsymbol{X}}^{\overline{x}_0,\overline{U}} = \mathscr{A}\overline{x}_0 + H\overline{U} + G\overline{\mu}_{\boldsymbol{W}}, \quad C_{\boldsymbol{X}} = GC_{\boldsymbol{W}}G^\top. \tag{5.13b}$$

In contrast to the affine disturbance feedback controller synthesis, open-loop controller synthesis can be accommodate hard control bounds. Specifically, instead of a chance-constraint relaxation considered in (4.17), we consider the following optimization problem (equivalent to (4.14)),

$$\underset{\overline{U}}{\text{minimize}} \quad \mathbb{P}_{\boldsymbol{X}}^{\overline{x}_0,\overline{U}}\{\boldsymbol{X} \in \mathscr{T}\} \tag{5.14a}$$

$$\text{subject to} \quad (5.13), \quad \overline{U} \in \mathcal{U}^N, \quad \overline{x}_0 \in \mathcal{T}_0 \tag{5.14b}$$

Recall that (5.14) is a log-concave optimization problem (Theorem 9b). The optimal value of (5.14), denoted by the value function $W_0^*(\overline{x}) : \mathcal{X} \to [0,1]$, serves as a lower bound to the maximal reach probability (4.1) by Theorem 10.

5.6.1 Fourier Transforms-Based Approach

By (5.13) and Assumption 8, we know that (5.14a) is the integral of a multivariate Gaussian over a polytope. Additionally, by (5.13b), the open-loop controller $\overline{U}$ influences only $\overline{\mu}_{\boldsymbol{X}}^{\overline{x}_0,\overline{U}}$, the mean of $\boldsymbol{X}$. Therefore, an approach to solve (5.14) would be to perform a nonlinear optimization of (5.14a), which in turn may be evaluated by numerical integration.

To compute (5.14a), we use Genz's algorithm [Gen14], which is based on quasi-Monte-Carlo simulations and Cholesky decomposition [Gen92]. Genz's algorithm provides an error estimate that is the result of a trade-off between accuracy and

computation time. We set the number of particles for the Monte-Carlo simulation so that the error estimate is less than some $\epsilon_{\text{genz}} > 0$. This results in a runtime evaluation of $\mathbb{P}_{\boldsymbol{X}}^{\bar{x}_0,\overline{U}}\{\boldsymbol{X} \in \mathscr{T}\}$ that is dependent on $\bar{x}_0$, unlike typical Monte-Carlo simulations. Additionally, we optimize the logarithm of $\mathbb{P}_{\boldsymbol{X}}^{\bar{x}_0,\overline{U}}\{\boldsymbol{X} \in \mathscr{T}\}$ instead of (5.14a), due to the log-concavity of the problem. Therefore, we set $\mathbb{P}_{\boldsymbol{X}}^{\bar{x}_0,\overline{U}}\{\boldsymbol{X} \in \mathscr{T}\} = \epsilon_{\text{genz}}$ if $\mathbb{P}_{\boldsymbol{X}}^{\bar{x}_0,\overline{U}}\{\boldsymbol{X} \in \mathscr{T}\} < \epsilon_{\text{genz}}$.

While the convexity of the logarithm-based reformulation ensures a tractable, globally optimal solution to (5.14), the lack of a closed-form expression for the objective (5.14a) requires black-box optimization techniques. Further, since Genz's algorithm enforces an accuracy of only ϵ_{genz}, the log-concavity of $\mathbb{P}_{\boldsymbol{X}}^{\bar{x}_0,\overline{U}}\{\boldsymbol{X} \in \mathscr{T}\}$ may not be preserved in the implementation. Hence the ideal solver for Problem B should handle the "noisy" evaluation of (5.14a) as an oracle, and solve a constrained optimization problem.

While this approach is significantly simplified for a Gaussian disturbance, we can in general utilize Fourier transforms (see Section 3.4.2) to compute the objective (5.14a). Note that log-concavity of (5.14) extends to any disturbance with a log-concave probability density function (Theorem 9b).

We use MATLAB's `patternsearch` to solve Problem (5.14), because it is based on direct search optimization [KLT03] and can handle estimation errors in (5.14a) efficiently. The solver is a derivative-free optimizer and uses evaluations over an adaptive mesh to obtain feasible descents towards the globally optimal solution. However, it requires a larger number of function evaluations as compared to MATLAB's `fmincon`.

Advantages and Limitations of this Approach

The main advantage of this approach is that it does not require gridding of the state, input, or disturbance spaces. Additionally, due to its focus on $\overline{x}_0$, it can perform safety analysis for an initial state of interest as opposed to the forced evaluation of

safety of all the points in the state space like dynamic programming [Aba+07]. By converting the stochastic reachability problem into an optimization problem involving a multi-dimensional integral, this approach achieves higher computational speed at lower memory cost for a given initial condition, which enables scalability.

While evaluating (5.14a) can be computationally expensive for arbitrary disturbances, for Gaussian disturbances we can compute (5.14a) efficiently. Further, since the dimension of the integral in (5.14a) is nN, large n effectively limits the time horizon N. Additionally, the lack of feedback in the controller used in (5.14), we can not have a large time horizon N [LOE13].

5.6.2 Chance Constraints-Based Approach

Using risk allocation or quantile function-based reformulation (Section 5.5.1) and piecewise affine approximations (Section 5.5), we can reduce (5.14) to a simple linear program (5.15).

$$
\begin{aligned}
\text{minimize} \quad & \sum\nolimits_{i=1}^{L_X} \alpha_i \\
\text{subject to} \quad & \overline{\mu}_{\boldsymbol{X}}^{\overline{x}_0,\overline{U}} = \bar{A}\overline{x}_0 + H\overline{U} + E\overline{\mu}_{\boldsymbol{W}} \\
& P\overline{U} \leq \overline{q},\ \overline{\alpha} \in [0, 0.5]^{L_X},\ (5.10a),\ \sum\nolimits_{i=1}^{L_X} \alpha_1 \leq 1 \\
& \overline{g}_i^\top \overline{\mu}_{\boldsymbol{X}}^{\overline{x}_0,\overline{U}} + t_{\boldsymbol{X},i}\sqrt{\overline{g}_i^\top C_{\boldsymbol{X}}^{\overline{0}} \overline{g}_i} \leq h_i,\ \forall i \in \mathbb{N}_{[1,L_X]}
\end{aligned}
\tag{5.15}
$$

with decision variables $\overline{\mu}_{\boldsymbol{X}}^{\overline{x}_0,\overline{U}}, \overline{U}, \overline{\alpha}$, and $\bar{t}_{\boldsymbol{X}}$. A similar convex chance constraint based open-loop controller synthesis technique was proposed in [LOE13]. However, the authors did not utilize the quantile function-based reformulation. Instead, the authors used MATLAB's `fmincon` to solve their optimization problem which led to significant numerical challenges. On the other hand, the linear program formulation (5.15) provides a significant improvement in computational tractability for open-loop controller synthesis. Denoting the optimal risk allocation for (5.15) as $\overline{\alpha}^*$ and the associated

optimal open-loop controller as $\overline{U}^*$, we obtain an underapproximation of the maximal reach probability $W_0(\overline{x}_0, \overline{U}) \geq 1 - \sum_{i=1}^{L_X} \alpha_i^*$ due to Boole's inequality (5.7). Thus,

$$r_{\overline{x}_0}^{\pi^*}(\mathscr{T})(\overline{x}_0) \geq 1 - \sum\nolimits_{i=1}^{L_X} \alpha_i^*, \ \forall \overline{x}_0 \in \mathcal{X}. \tag{5.16}$$

Advantages and Limitations of this Approach

Similar to the Fourier transform approach (Section 5.6.1), the chance constraints-based approach is grid-free and scales well. Since it is a linear program, this method scales much better than the Fourier transform-based approach. Additionally, being a point-based evaluation of stochastic reachability, it significantly outperforms the dynamic programming in computation cost and memory when the safety of a given initial state is of interest.

The Fourier transform-based approach can be interpreted as a joint chance constraint optimization approach, while (5.15) is an individual chance constraint-based restriction. This restriction, derived via Boole's inequality (5.7), can become overly conservative under high violation probabilities of the individual hyperplanes $\overline{g}_i^\top \boldsymbol{X} \leq h_i$.

5.6.3 Sampling-Based Approach

In [LOE13], a particle control approach was proposed to solve the following mixed-integer program to obtain an approximation to (4.14),

$$\begin{aligned}
\underset{\overline{z},\overline{X},\overline{U}}{\text{maximize}} \quad & \textstyle\sum_{i=1}^{M_{\text{particles}}} z_i \\
\text{subject to} \quad & \overline{z} \in \{0,1\}^{M_{\text{particles}}} \\
& \overline{U} \in \mathcal{U}^N \\
& \overline{X}^{(i)} = \mathscr{A}\overline{x}_0 + H\overline{U} + E\overline{W}^{(i)}, \quad \forall i \in \mathbb{N}_{[1,M_{\text{particles}}]} \\
& \overline{g}_j^\top \overline{X}^{(i)} - h_j \leq C(1 - z_i), \quad \forall i \in \mathbb{N}_{[1,M_{\text{particles}}]}, \forall j \in \mathbb{N}_{[1,L_X]}
\end{aligned} \tag{5.17}$$

where $M_{\text{particles}}$ is the user-specified number of scenarios considered, $\overline{W}^{(i)}$ with $i \in \mathbb{N}_{[1,M_{\text{particles}}]}$ are $M_{\text{particles}}$ realizations of the concatenated disturbance vector $\boldsymbol{W}$, and C is a large constant.

This approach was further improved upon in [Sar+19]. Specifically, a lower bound on the number of particles needed to bound the overapproximation error was obtained. This lower bound, based on Hoeffding's inequality, is independent of the stochasticity of the disturbance. Additionally, a Voronoi partition-based undersampling technique was developed to significantly improve the tractability of this approach using clustering techniques. The resulting simpler mixed-integer linear program underapproximates the maximal reach probability estimated by (5.17).

Advantages and Limitations of this Approach

In contrast to the Fourier transform-based and chance-constrained-based approaches (Section 5.6.1 and 5.6.2), sampling-based approach does not require Gaussianity assumption on the disturbance for tractability. It allows approximation of the maximal reach probability for linear systems with non-Gaussian disturbances. With a lower bound available on the number of particles needed to achieve a desired overapproximation, the quality of the solution is also user-controlled.

The major drawback of this approach is its reliance on the mixed-integer linear program formulation, which requires significant computational efforts in some cases [LOE13]. However, with the use of clustering techniques, tractable approximations are now possible.

5.7 Polytopic Underapproximation of Stochastic Reach Set using Open-loop Controllers

Recall that the stochastic reach set is underapproximated by the open-loop controller-based stochastic reach set, $\mathcal{K}_0^{\rho^*}(\alpha, \mathscr{T}) \subseteq \mathcal{L}_0^{\pi^*}(\alpha, \mathscr{T})$ (see Theorem 10). Given a finite

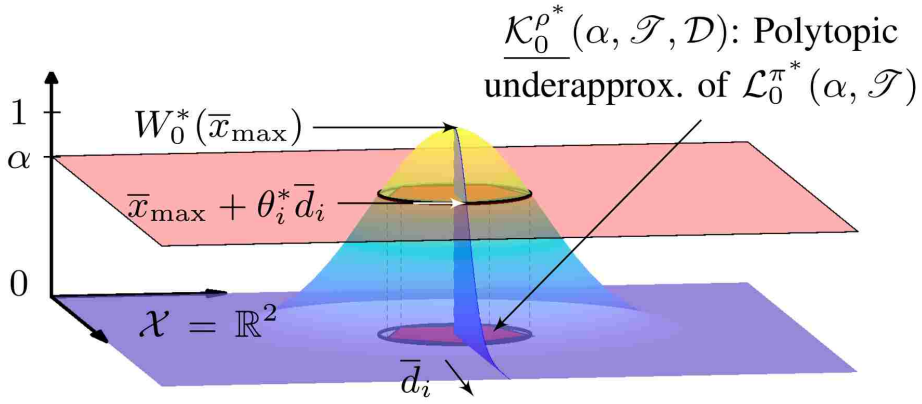


Figure 5.2: Components of Algorithm 6. We compute $W_0^*(\overline{x}_{\max})$ by solving (5.18) and then for each $\bar{d}_i \in \mathcal{D}$, compute vertices of $\underline{\mathcal{K}_0^{\rho^*}}(\alpha, \mathscr{T}, \mathcal{D})$ by solving (5.19), and construct $\underline{\mathcal{K}_0^{\rho^*}}(\alpha, \mathscr{T}, \mathcal{D})$ by computing the convex hull of the computed vertices.

set $\mathcal{D} \subset \mathcal{X}$ consisting of direction vectors $\bar{d}_i$, we propose Algorithm 6 to compute a tight polytopic underapproximation of $\mathcal{K}_0^{\rho^*}(\alpha, \mathscr{T})$ using the ray-shooting algorithm (Figure 2.1):

1. find $\overline{x}_{\max} = \arg\max_{\overline{x} \in \mathcal{X}} W_0^*(\overline{x})$; if $W_0^*(\overline{x}_{\max}) < \alpha$, then $\mathcal{K}_0^{\rho^*}(\alpha, \mathscr{T}) = \emptyset$; else, continue,

2. obtain relative boundary points of the set $\mathcal{K}_0^{\rho^*}(\alpha, \mathscr{T})$ via line searches from $\overline{x}_{\max}$ along the directions $\bar{d}_i \in \mathcal{D}$, and

3. compute the convex hull of the computed relative boundary points to obtain a polytope $\underline{\mathcal{K}_0^{\rho^*}}(\alpha, \mathscr{T}, \mathcal{D})$.

Figure 5.2 illustrates these steps. By [BV04, Sec. 2.1.4], we have $\underline{\mathcal{K}_0^{\rho^*}}(\alpha, \mathscr{T}, \mathcal{D}) \subseteq \mathcal{K}_0^{\rho^*}(\alpha, \mathscr{T})$. We have equality when all of the extreme points of $\mathcal{K}_0^{\rho^*}(\alpha, \mathscr{T})$ are discovered by this approach (possible for a polytopic $\mathcal{K}_0^{\rho^*}(\alpha, \mathscr{T})$) [Web94, Thm. 2.6.16]. The steps 2) and 3) are enabled by the compactness and convexity of $\mathcal{K}_0^{\rho^*}(\alpha, \mathscr{T})$ [VO18b, Prop. 4] [Web94, Ch. 2].

Compute $\overline{x}_{\max}$ that Maximizes $W_0^*(\overline{x})$

We solve the following optimization problem

$$\begin{aligned} &\underset{\overline{x},\overline{U},\beta}{\text{maximize}} \quad \beta \\ &\text{subject to} \quad \begin{cases} \overline{x} \in \mathcal{T}_0, \quad \overline{U} \in \mathcal{U}^N, \quad \beta \in [0,1], \\ W_0(\overline{x},\overline{U}) \geq \beta, \quad \beta \geq \alpha \end{cases} \end{aligned} \tag{5.18}$$

We denote the optimal solution of (5.18) as $\overline{x}_{\max} \in \mathcal{X}$ (the maximizer of $W_0^*(\overline{x})$), the associated optimal open-loop controller $\overline{U}_{\max} \in \mathcal{U}^N$, and the highest value of maximal reach probability $\beta^* = W_0^*(\overline{x}_{\max})$ with $W_0^*(\overline{x})$ given by (4.15c) at $k = 0$. The optimization problem (5.18) is (4.15c) written in the epigraph form [BV04, Eq. 4.11], with an additional constraint of $W_0(\overline{x},\overline{U})) \geq \alpha$.

By Theorem 9b, applying $\log(\cdot)$ to the constraint $W_0(\overline{x},\overline{U}) \geq \beta$ converts (5.18) into a convex problem. The formulation of (5.18) ensures that if it is infeasible, then $\underline{\mathcal{K}_0^{\rho^*}}(\alpha,\mathscr{T},\mathcal{D})$ and $\mathcal{K}_0^{\rho^*}(\alpha,\mathscr{T})$ are empty. We cannot conclude that $\mathcal{L}_0^{\pi^*}(\alpha,\mathscr{T})$ is empty, because of the underapproximative nature of our approach (Theorem 10).

Compute Relative Boundary Points of $\mathcal{K}_0^{\rho^*}(\alpha,\mathscr{T})$ via Line Searches

To compute the relative boundary points of $\mathcal{K}_0^{\rho^*}(\alpha,\mathscr{T})$, we must solve for each $i \in \mathbb{N}_{[1,|\mathcal{D}|]}$

$$\begin{aligned} &\underset{\theta_i,\overline{U}_i}{\text{maximize}} \quad \theta_i \\ &\text{subject to} \quad \begin{cases} \overline{U}_i \in \mathcal{U}^N, \quad \theta_i \in \mathbb{R}, \quad \theta_i \geq 0 \\ W_0(\overline{x}_{\max} + \theta_i \bar{d}_i, \overline{U}_i) \geq \alpha \end{cases} \cdot \end{aligned} \tag{5.19}$$

We denote the optimal solution of (5.19) as θ_i^* and $\overline{U}_i^*$. The optimization problem (5.19) is also known as a line search problem, an integral component of the gradient descent algorithms [BV04, Sec. 9.3]. It may be solved exactly via convex optimization

(by Theorem 9b and [BV04, Sec. 3.2.2]), or approximatively via backtracking line search [BV04, Pg. 465]. Note that, by construction, $W_0(\overline{x}_{\max} + \theta_i^* \bar{d}_i, \overline{U}_i^*) = \alpha$, and for any $\epsilon > 0$, $W_0(\overline{x}_{\max} + (\theta_i^* + \epsilon)\bar{d}_i, \overline{U}_i^*) < \alpha$. Hence, $\overline{x}_{\max} + \theta_i^* \bar{d}_i \in \partial \mathcal{K}_0^{\rho^*}(\alpha, \mathscr{T})$, and the optimal open-loop controller from this relative boundary point is $\rho^*(\overline{x}_{\max} + \theta_i^* \bar{d}_i) = \overline{U}_i^*$ [VO18b, Prop. 4].

Construction of $\underline{\mathcal{K}_0^{\rho^*}}(\alpha, \mathscr{T}, \mathcal{D})$

If (5.18) has a solution, we have $W_0^*(\overline{x}_{\max}) \geq \alpha$, and we construct the polytope $\underline{\mathcal{K}_0^{\rho^*}}(\alpha, \mathscr{T}, \mathcal{D})$ via the convex hull of the computed relative boundary points $\overline{x}_{\max} + \theta_i^* \bar{d}_i$, $\forall i \in \mathbb{N}_{[1,|\mathcal{D}|]}$. Since $\mathcal{K}_0^{\rho^*}(\alpha, \mathscr{T})$ is convex and compact, and $\overline{x}_{\max} + \theta_i^* \bar{d}_i \in \partial \mathcal{K}_0^{\rho^*}(\alpha, \mathscr{T})$, we have $\underline{\mathcal{K}_0^{\rho^*}}(\alpha, \mathscr{T}, \mathcal{D}) \subseteq \mathcal{K}_0^{\rho^*}(\alpha, \mathscr{T})$ [BV04, Sec. 2.1.4]. On the other hand, if (5.18) is infeasible, then $\mathcal{K}_0^{\rho^*}(\alpha, \mathscr{T})$ is empty, which implies $\underline{\mathcal{K}_0^{\rho^*}}(\alpha, \mathscr{T}, \mathcal{D})$ is empty.

Algorithm 6 Compute polytopic $\underline{\mathcal{K}_0^{\rho^*}}(\alpha, \mathscr{T}, \mathcal{D}) \subseteq \mathcal{L}_0^{\pi^*}(\alpha, \mathscr{T})$

Input: Probability threshold α, set of direction vectors $\mathcal{D}$
Output: $\underline{\mathcal{K}_0^{\rho^*}}(\alpha, \mathscr{T}, \mathcal{D}) \subseteq \mathcal{K}_0^{\rho^*}(\alpha, \mathscr{T}) \subseteq \mathcal{L}_0^{\pi^*}(\alpha, \mathscr{T})$

```
Solve (5.18) to compute x̄_max
if W_0^*(x̄_max) ≥ α then
    for d̄_i ∈ D do
        Solve (5.19) to compute a relative boundary point x̄_max + θ_i^* d̄_i and its optimal open-loop controller ρ*(x̄_max + θ_i^* d̄_i)
    end for
    K_0^ρ*(α, T, D) ← conv_{i∈N_[1,|D|]}(x̄_max + θ_i^* d̄_i)
else
    K_0^ρ*(α, T, D) ← ∅
end if
```

Algorithm 6 solves Problem 9.

5.7.1 Implementational Details and Computational Effort

Algorithm 6 is an anytime algorithm, as interrupting the convex hull of the solutions of (5.19) for an arbitrary subset of direction vectors in $\mathcal{D}$ also provides a valid un-

derapproximation. Additionally, Algorithm 6 is parallelizable since the computations along each of the direction $\bar{d}_i$ are independent.

The choice of $\mathcal{D}$ influences the quality (in terms of volume) of underapproximation provided by Algorithm 6. One strategy is to choose the vectors in $\mathcal{D}$ that are spaced far apart initially, and then increase $|\mathcal{D}|$ by sampling appropriate directions to tighten the underapproximation as appropriate, at the cost of increased computational time.

The computation of $\overline{x}_{\max}$ also determines the quality of the polytopic underapproximation. Motivated by the Chebyshev centering of a polytope [BV04, Sec. 8.5.1], we consider an alternative to (5.18),

$$\begin{aligned} &\underset{\overline{x},\overline{U},R}{\text{maximize}} \quad R \\ &\text{subject to} \quad \begin{cases} \overline{U} \in \mathcal{U}^N, \quad \beta \in [0,1], \quad W_0(\overline{x},\overline{U}) \geq \alpha \\ R \geq 0, \quad \overline{a}_i^\top \overline{x} + R\|\overline{a}_i\|_2 \leq b_i, \forall i \in \mathbb{N}_{[1,M]} \end{cases} . \end{aligned} \tag{5.20}$$

Here, instead of finding the initial state with maximum $W_0(\overline{x},\overline{U})$, we wish to compute the "deepest" point in the set $\mathcal{T}_0$ such that $W_0(\overline{x},\overline{U}) \geq \alpha$. Thus, the requirement that $\overline{x} \in \mathcal{T}_0$ is tightened to be $\mathcal{T}_0 = \cap_{i=1}^M \{\overline{z} : \overline{a}_i^\top \overline{z} \leq b_i\}$. We also have a feasibility problem in $\overline{x}$ and $\overline{U}$. Due to the convexity of the underapproximative polytope, one can combine both of these polytopes (via convex hull of their vertices) to obtain a larger polytope. Larger R suggests that considering a collection of direction vectors $\mathcal{D}$, sampled from a unit circle is not a bad choice.

Memory Requirements

The memory requirements of Algorithm 6 grow linearly with $|\mathcal{D}|$ and are independent of the system dimension. In contrast, dynamic programming requires an exponential number of grid points in memory, leading to the curse of dimensionality [Aba+07]. Algorithm 6 is grid-free and recursion-free, and it scales favorably with the system dimension, as compared to dynamic programming. Thus, Algorithm 6 can verify and

synthesize controllers for high-dimensional systems that were previously not verifiable.

Computation Time

Denoting the computation times to solve (5.18) and (5.19) as t_{xmax} and t_{line} respectively, the computation time for Algorithm 6 is $\mathcal{O}(t_{\text{xmax}} + t_{\text{line}}|\mathcal{D}|)$. Since (5.18) and (5.19) are convex problems, globally optimal solutions are assured with (potentially) low t_{xmax} and t_{line}. However, the joint chance constraint $W_0(\overline{x}, \overline{U}) \geq \alpha$ is not solver-friendly, since we do not have a closed-form expression for $W_0(\overline{x}, \overline{U})$, or an exact reformulation into a conic constraint.

In Algorithm 6, to solve (5.18) and (5.19), we need an efficient way to enforce the constraint $W_0(\overline{x}, \overline{U}) \geq \alpha$ for any $\overline{x} \in \mathcal{X}$, $\overline{U} \in \mathcal{U}^N$ and $\alpha \in [0, 1]$. Under Assumption 8, $W_0(\overline{x}_0, \overline{U})$ is the integration of a Gaussian PDF over a polytope [VO17]. We consider three approaches to enforce the constraint $W_0(\overline{x}_0, \overline{U}) \geq \alpha$, motivated by the discussion in Section 5.6.

1. Convex chance constraints: An underapproximative reformulation via Boole's inequality and Gaussian random vector properties [LOE13]. A sufficient condition for convexity of this reformulation requires $\alpha \in [0.5, 1]$ (see [LOE13; OW08]).

2. Sampling: Mixed integer-linear reformulation via scenarios drawn from $\mathbb{P}_{\boldsymbol{X}}^{\overline{x}_0, \overline{U}}$ that *optionally* satisfy the reach objective (stay within the target tube) [LOE13].

3. Fourier transform: An approximative reformulation via a numerical "noisy" Monte Carlo simulations-based evaluation of $W_0(\overline{x}_0, \overline{U})$. We rely on gradient-free optimization techniques [KLT03] to optimize the resulting "noisy" optimization problem [VO17; VO18b]. For a Gaussian $\boldsymbol{X}$ (2.27), we use Genz's algorithm to evaluate $W_0(\overline{x}, \overline{U})$ via quasi-Monte Carlo simulations and Cholesky decomposition [Gen14; Gen92].

Enforce $W_0(\overline{x}, \overline{U}) \geq \alpha$	Approximation	Implementation
Convex chance constraints [LOE13; OW08]	Tighten the constraint $W_0(\overline{x}, \overline{U}) \geq \alpha$ via Boole's inequality [LOE13]	Nonlinear solver [LOE13] or iterative linear programs [OW08]; Requires $\alpha \geq 0.5$ for convexity of $\{\overline{U} : W_0(\overline{x}_0, \overline{U}) \geq \alpha\}$
Sampling-based [LOE13; BOW11]	Approximation quality improves with increase in the number of samples N_s [LOE13]	Mixed-integer linear program [LOE13; BOW11]; Exponential compute cost w.r.t. N_s [BOW11]
Fourier transform (Evaluate $W_0(\overline{x}, \overline{U})$ numerically) [VO17; VO18b]	Approximation to a desired tolerance [VO17; VO18b]	Nonlinear solver that can handle noisy objectives [KLT03]; Use Genz's algorithm [Gen92] to evaluate $W_0(\overline{x}, \overline{U})$

Table 5.1: Enforcing $W_0(\overline{x}_0, \overline{U}) \geq \alpha$ under Assumption 8.

Table 5.1 compares the implementation of the constraint $W_0(\overline{x}, \overline{U}) \geq \alpha$ using these approaches. Figure 5.3 summarizes the conservativeness introduced at different stages of the open-loop controller-based underapproximation when chance constrained reformulation is used.

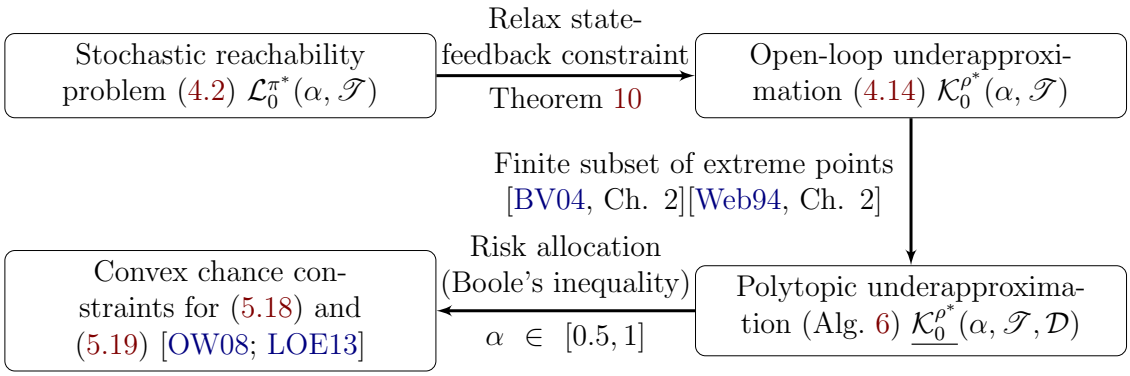


Figure 5.3: Construction of polytopic underapproximation of the stochastic reach set, $\underline{\mathcal{K}_0^{\rho^*}}(\alpha, \mathscr{T}, \mathcal{D}) \subseteq \mathcal{L}_0^{\pi^*}(\alpha, \mathscr{T})$, via chance constraints and open-loop controllers.

5.8 Numerical Experiments

All computations were performed using MATLAB on an Intel Xeon CPU with 3.4GHz clock rate and 32 GB RAM. We used `SReachTools` (see Chapter 6) for the stochastic reachability problems, `MPT3` [Her+13] for computational geometry, and `CVX` [GB14] for parsing convex problems. We used `Gurobi` [Gur15] as the backend solver for the chance constrained approach, and MATLAB's `patternsearch` as the nonlinear solver for the Fourier transform approach. We will demonstrate various features of `SReachTools` using these experiments (see Table 6.1, pg. 145).

5.8.1 Demonstration of Interpolation Technique and Scalability on a Chain of Integrators

Consider a chain of integrators,

$$\boldsymbol{x}_{k+1} = \begin{bmatrix} 1 & N_s & \frac{1}{2}N_s^2 & \dots & \frac{1}{(n-1)!}N_s^{n-1} \\ 0 & 1 & N_s & & \\ \vdots & & & \ddots & \vdots \\ 0 & 0 & 0 & \dots & N_s \\ 0 & 0 & 0 & \dots & 1 \end{bmatrix} \boldsymbol{x}_k + \begin{bmatrix} \frac{1}{n!}N_s^n & \dots & \frac{1}{2}N_s & N_s \end{bmatrix}^\top u_k + \boldsymbol{w}_k \tag{5.21}$$

with state $\boldsymbol{x}_k \in \mathbb{R}^n$, input $u_k \in \mathcal{U} = [-1, 1]$, a Gaussian disturbance $\boldsymbol{w}_k \sim \mathcal{N}(\overline{0}_2, 0.01 I_2)$, sampling time $N_s = 0.1$, and time horizon $N = 5$. Here, I_n refers to the n-dimensional identity matrix and $\overline{0}_n$ is the n-dimensional zero vector.

2D system

Consider the terminal time problem with $\mathcal{T}_i = [-1, 1]^2$ $i \in \mathbb{N}_{[0,N-1]}$ and $\mathcal{T}_N = [-0.5, 0.5]^2$. We compare $\underline{\mathcal{K}_0^{\rho^*}}(\alpha, \mathscr{T}, \mathcal{D})$ obtained using Algorithm 6 with $|\mathcal{D}| = 32$

	$n = 2$ ($\lvert\mathcal{D}\rvert = 32$)			$n = 40$ ($\lvert\mathcal{D}\rvert = 6$)		
α	0.6	0.85	0.9	0.6	0.85	0.9
Algorithm 6	19.36	19.28	18.94	6138.7	6101.2	6101.2
Interpolate $\mathcal{K}_0^{\rho^*}(\alpha, \mathscr{T})$	–	0.037	–	–	0.06	–
Dynamic programming	47.37			Not possible		
Interpolate $\mathcal{L}_0^{\pi^*}(\alpha, \mathscr{T})$	–	0.04	–			

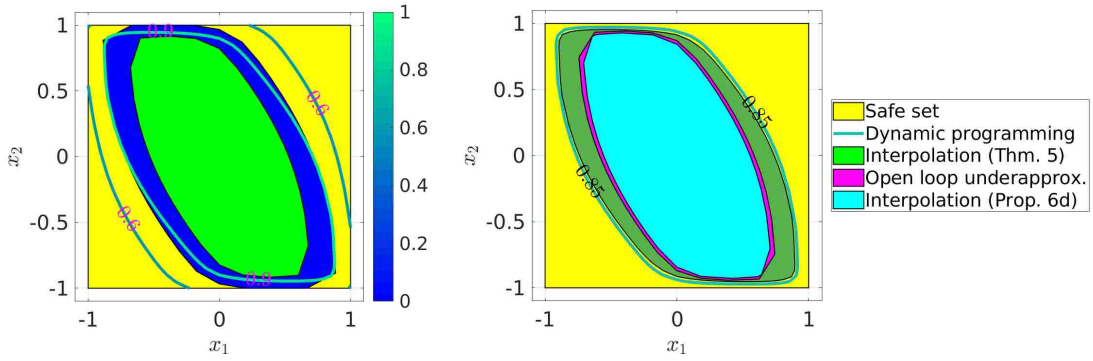
Table 5.2: Computation time (in seconds) for verification of a chain of integrators.

and the set $\mathcal{L}_0^{\pi^*}(\alpha, \mathscr{T})$ obtained via grid-based dynamic programming [Aba+07]. We discretized the state space and the input space in steps of 0.05.

Figure 5.4a shows that, in general, Algorithm 6 provides a good underapproximation of the true stochastic viable set for a double integrator. The advantage of using state-feedback π^* over an open loop controller ρ^* is seen in the underapproximation "gaps" between the polytopes (Theorem 10). Figure 5.4b shows that the (interpolated) polytopic underapproximation obtained at $\alpha = 0.85$ using the polytopic representations of $\mathcal{L}_0^{\pi^*}(\alpha, \mathscr{T})$ and $\mathcal{K}_0^{\rho^*}(\alpha, \mathscr{T})$ for $\alpha \in \{0.6, 0.9\}$ (Theorems 8 and 9d) approximates the true sets well.

Table 5.2 provides the computation times. As expected, the grid-free nature of Algorithm 6 coupled with the convexity and compactness properties established in Section 4.6 and the underapproximative guarantee (Theorem 10) provides significant speed-up for underapproximative verification and controller synthesis. The interpolation approach (Theorems 8 and 9d) took significantly lower computation time while producing a good underapproximation. We can now perform real-time verification by computing a few stochastic reach sets offline and then performing appropriate interpolations.

For Figure 5.5, we chose $\mathcal{T}_i = [-1, 1]^2$ (viability problem), $\mathcal{U} = [-0.1, 0.1]$, $\boldsymbol{w}_k \sim \mathcal{N}(\bar{0}_{2\times 1}, \mathrm{diag}([10^{-6}\ 10^{-3}])$, sampling time $N_s = 0.25$, and time horizon $N = 5$. We chose 24 direction vectors, sampled from a unit circle such that they subtend the same angle at the center. For `SReachDynProg`, we discretized the state and input



a) b)

Figure 5.4: Stochastic reachability for a double integrator: a) stochastic viable sets $\mathcal{L}_0^{\pi^*}(\alpha, \mathscr{T})$ (contours) and their polytopic underapproximations $\underline{\mathcal{K}_0^{\rho^*}}(\alpha, \mathscr{T}, \mathcal{D})$ for $\alpha \in \{0.6, 0.9\}$; b) the true sets ($\mathcal{L}_0^{\pi^*}(\alpha, \mathscr{T})$ and $\underline{\mathcal{K}_0^{\rho^*}}(\alpha, \mathscr{T}, \mathcal{D})$) at $\alpha = 0.85$ and their underapproximative interpolations (Thms. 8 and 9d) using their counterparts for $\alpha \in \{0.6, 0.9\}$.

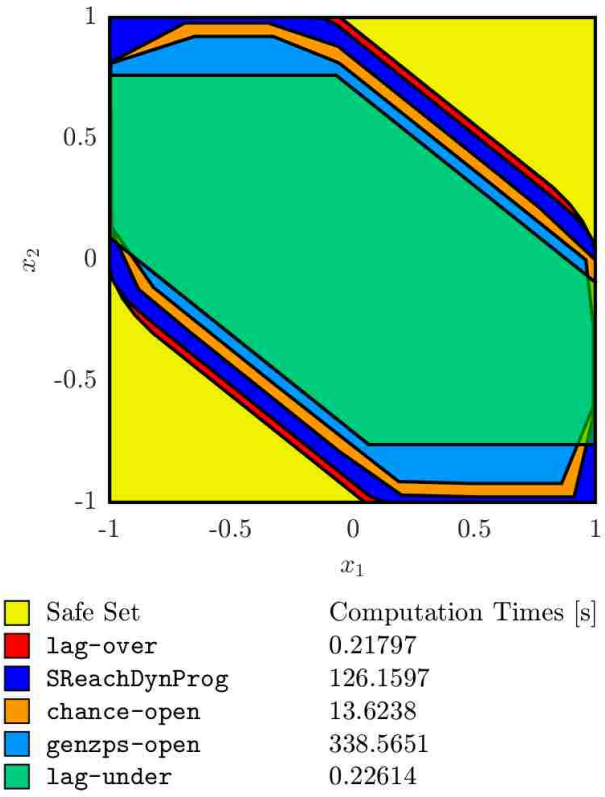


Figure 5.5: Comparison of $\mathcal{L}_0^{\pi^*}(0.8)$ for a 2-dimensional double integrator computed via dynamic programming, and multiple `SReachSet` approximations.

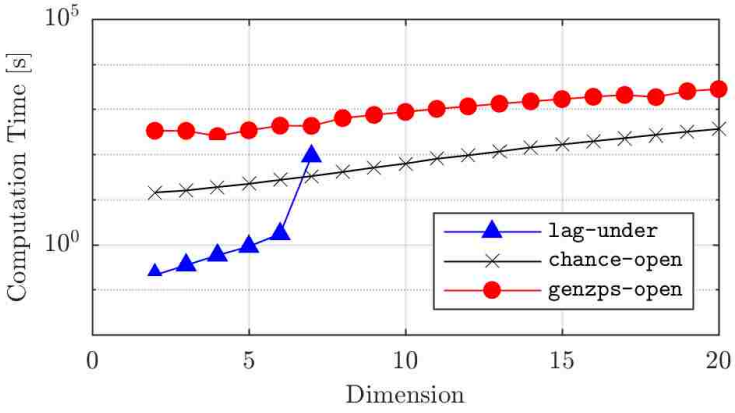


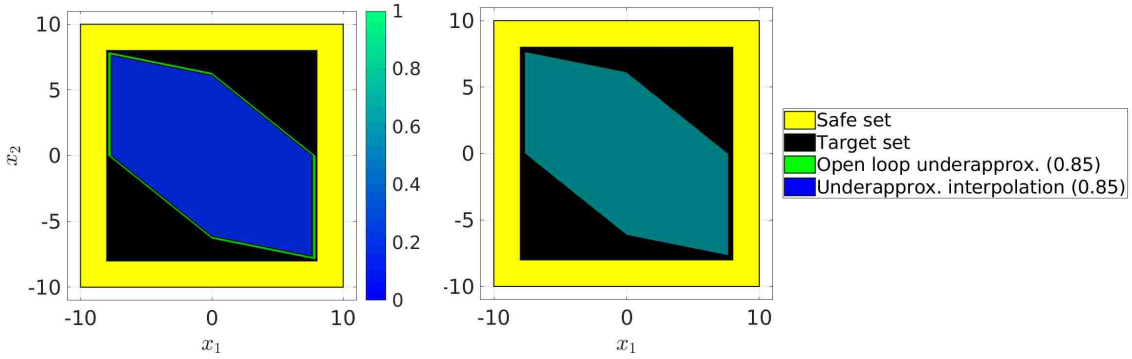
Figure 5.6: Scalability of `SReachSet` methods demonstrated on a chain of integrators, as state dimension increases.

spaces using a grid step sizes of 0.02 and 0.01 respectively. As expected, Algorithm 6 provides an underapproximation to the stochastic reach set $\mathcal{L}_0^{\pi^*}(\alpha, \mathscr{T})$.

For Figure 5.6, at each iteration of the state dimension $n \in \mathbb{N}_{[2,20]}$ we chose $\mathcal{T}_i = [-1, 1]^n$ (viability problem), $\mathcal{U} = [-0.1, 0.1]$, $\boldsymbol{w}_k \sim \mathcal{N}(\overline{0}_{n\times 1}, \mathrm{diag}([10^{-6} \ \ldots \ 10^{-6} \ 10^{-3}])$, sampling time $N_s = 0.25$, and time horizon $N = 5$. We chose 24 direction vectors, sampled from a unit circle such that they subtend the same angle at the center. Lagrangian methods are fast in low dimensions, but practically limited to $m \leq 7$ because of the vertex-facet enumeration problem. Open-loop methods, `genzps-open` and `chance-open`, scale well with dimension, but have a higher initial cost.

40D System

To demonstrate scalability, consider the terminal time problem with $n = 40$ and $\mathcal{T}_i = [-10, 10]^{40}$ $i \in \mathbb{N}_{[0,N-1]}$ and $\mathcal{T}_N = [-8, 8]^{40}$. Due to the curse of dimensionality, we can not use dynamic programming [Aba+07]. We compute $\underline{\mathcal{K}_0^{\rho^*}}(\alpha, \mathscr{T}, \mathcal{D})$ for $n = 40$ for $\alpha \in \{0.6, 0.85, 0.9\}$ and also demonstrate the underapproximative interpolation for this high-dimensional system. Figure 5.7a shows a 2D slice of $\underline{\mathcal{K}_0^{\rho^*}}(\alpha, \mathscr{T}, \mathcal{D})$ that verifies $\overline{x}_0$ of the form $[x_1 \ x_2 \ x_3 \ 0 \ \ldots \ 0]^\top \in \mathbb{R}^{40}$. The difference in volume between



a) b)

Figure 5.7: Stochastic reach-avoid analysis for a chain of integrators ($n = 40$): a) the polytopic underapproximations of the stochastic reach-avoid sets $\underline{\mathcal{K}_0^{\rho^*}}(\alpha, \mathscr{T}, \mathcal{D})$ for $\alpha \in \{0.6, 0.9\}$; b) the $\underline{\mathcal{K}_0^{\rho^*}}(\alpha, \mathscr{T}, \mathcal{D})$ at $\alpha = 0.85$ and its tight underapproximative interpolation (Thm. 9d) using $\underline{\mathcal{K}_0^{\rho^*}}(\alpha, \mathscr{T}, \mathcal{D})$ for $\alpha \in \{0.6, 0.9\}$. For illustration, we consider initial states $\overline{x}_0 = [x_1 \; x_2 \; 0 \; \dots 0]^\top$.

the underapproximative interpolation (Theorem 9d) and $\underline{\mathcal{K}_0^{\rho^*}}(\alpha, \mathscr{T}, \mathcal{D})$ at $\alpha = 0.85$ is negligible (1.124 via MPT3 [Her+13]), as seen in Figure 5.7b. To the best of our knowledge, this is the largest stochastic LTI system verified to date through a stochastic reachability formulation.

5.8.2 Application: Spacecraft Rendezvous Problem

We consider two spacecraft in the same elliptical orbit. One spacecraft, referred to as the deputy, must approach and dock with another spacecraft, referred to as the chief, while remaining in a line-of-sight cone, in which accurate sensing of the other vehicle is possible. The relative dynamics are described by the Clohessy-Wiltshire-Hill (CWH) equations [Wie89] with additive stochastic noise to account for model uncertainties,

$$\ddot{x} - 3\omega x - 2\omega\dot{y} = m_d^{-1} F_x, \qquad \ddot{y} + 2\omega\dot{x} = m_d^{-1} F_y. \tag{5.22}$$

The chief is located at the origin, the position of the deputy is $x, y \in \mathbb{R}$, $\omega = \sqrt{\mu/R_0^3}$ is the orbital frequency, μ is the gravitational constant, and R_0 is the orbital radius of the spacecraft. See [LOE13; GVO17] for further details.

We define the state as $z = [x, y, \dot{x}, \dot{y}] \in \mathbb{R}^4$ and input as $u = [F_x, F_y] \in \mathcal{U} \subseteq \mathbb{R}^2$. We discretize the dynamics (5.22) in time to obtain the discrete-time LTI system,

$$z_{k+1} = Az_k + Bu_k + w_k \tag{5.23}$$

with $w_k \in \mathbb{R}^4$ a Gaussian i.i.d. disturbance, with $\mathbb{E}[w_k] = 0$, $\Sigma = \mathbb{E}[w_k w_k^\top] = 10^{-4} \times \text{diag}(1, 1, 5 \times 10^{-4}, 5 \times 10^{-4})$. Given $u_M \in \mathbb{R}, u_M > 0$, we define the input space as $\mathcal{U} = [-u_M, u_M]^2$ and the target tube for a time horizon of $N = 5$,

$$\begin{aligned}\mathcal{T}_5 = \big\{ z \in \mathbb{R}^4 : |z_1| \leq 0.1, -0.1 \leq z_2 \leq 0, \\ |z_3| \leq 0.01, |z_4| \leq 0.01 \big\}, \text{ and} \\ \mathcal{T}_i = \big\{ z \in \mathbb{R}^4 : |z_1| \leq z_2, \max(|z_3|, |z_4|) \leq 5u_M \big\}, i \in \mathbb{N}_{[0,4]}.\end{aligned}$$

We consider two verification (terminal time) problems as done in [VO18b; GVO17; LOE13]:

P1) initial velocity $\dot{x} = \dot{y} = 0$ km/s and input bound $u_M = 0.1$,

P2) initial velocity $\dot{x} = \dot{y} = 0.01$ km/s and input bound $u_M = 0.01$.

We are interested in solving these stochastic reach-avoid problems, by computing $\mathcal{K}_0^{\rho^*}(\alpha, \mathscr{T})$, at $\alpha = 0.8$.

We solve the terminal time problem conservatively using Algorithm 6 (via Fourier transform and chance constraint approaches) and Lagrangian methods [GVO17]. Both of these problems are intractable via dynamic programming [SL10]. Exploiting the convexity and compactness results from Section 4.6, Algorithm 6 performs significantly faster than the grid-based implementation of chance constraints approach

Underapproximative method	Chance constraint [OW08; LOE13]	Fourier transform [VO17; VO18b]	Lagrangian [GVO17]
Figure 5.8 ($\dot{x} = \dot{y} = 0$, $u_M = 0.1$)	16.87 ($\lvert\mathcal{D}\rvert = 32$)	940.45 ($\lvert\mathcal{D}\rvert = 8$)	14.5
Figure 5.9 ($\dot{x} = \dot{y} = u_M = 0.01$)	18.24 ($\lvert\mathcal{D}\rvert = 32$)	2029.51 ($\lvert\mathcal{D}\rvert = 8$)	–

Table 5.3: Computation times (in seconds) for verification of the satellite rendezvous problem. Dynamic programming [Aba+07] is not possible for 4D systems.

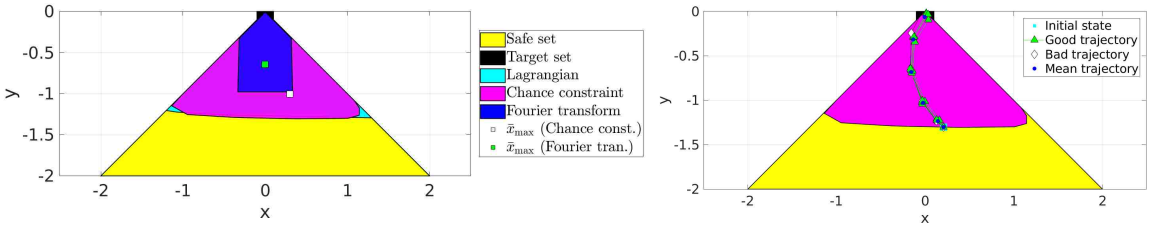


Figure 5.8: Underapproximative verification and open-loop controller synthesis for spacecraft rendezvous problem for zero initial velocity. Monte Carlo simulations (10^5 scenarios with five randomly chosen trajectories displayed) show a simulated reach probability of 0.82; the chance constraint estimate was 0.8.

proposed in [LOE13]. Figures 5.8 and 5.9 show a slice of the stochastic reach-avoid underapproximations for both the verification problems. Computational times are summarized in Table 5.3.

The Lagrangian method [GVO17] utilizes computational geometry to underapproximate the stochastic reach-avoid set and searches in the space of closed-loop controllers. However, it relies on the vertex-facet enumeration problem, which fails

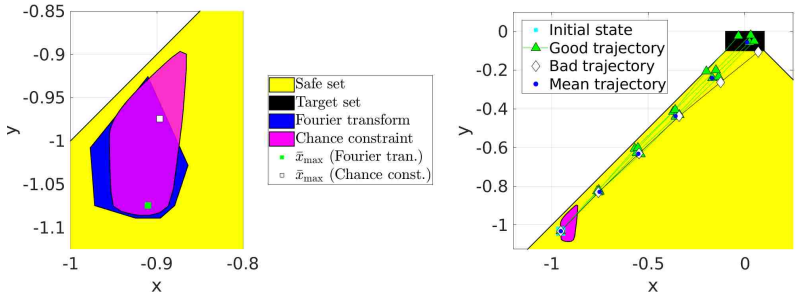


Figure 5.9: Underapproximative verification and open-loop controller synthesis for spacecraft rendezvous problem for non-zero initial velocity. Monte Carlo simulations (10^5 scenarios with five randomly chosen trajectories displayed) show a simulated reach probability of 0.81; the chance constraint estimate was 0.8.

for large time horizons, small target sets, or small safe sets. While this approach fails to compute a set for P2, it is slightly faster than Algorithm 6 (see Table 5.3) for P1. The associated set subsumes the polytopes computed using Algorithm 6 since Lagrangian method searches for a closed-loop controller in a conservative manner.

Within Algorithm 6, the implementation using chance constraints (via risk allocation [LOE13; OW08]) outperforms the implementation via Fourier transforms (Genz's algorithm and MATLAB's `patternsearch` [VO17; Gen92; KLT03]) in terms of computational time. The computational efficiency of chance constraint approach results from its use of a series of linear programs [OW08]. The Fourier transform approach evaluates $W_0(\overline{x}, \overline{U})$ using Genz's algorithm (quasi-Monte Carlo simulation) and solves the problems (5.18) and (5.19) using MATLAB's `patternsearch` and bisection [VO18b]. The Fourier transform approach does not have additional conservativeness (due to Boole's inequality), as compared to chance constraint approach (see Figure 5.9). However, due to the noisy nature of the optimization problem, the line search in the Fourier transform approach may terminate prematurely (see Figure 5.8).

5.8.3 Demonstration of Stochastic Reachability of a Target Tube for a LTV System: Dubin's Vehicle with a Fixed Turn Rate

We consider the problem of driving a Dubin's vehicle under a known turning rate sequence while staying within a target tube. The linear time-varying dynamics with additive disturbance describing the position of the car is given by

$$\boldsymbol{x}_{k+1} = \boldsymbol{x}_k + \underbrace{\begin{bmatrix} T_s \cos(\theta_0 + T_s \sum_{i=0}^{k-1} \omega_i) \\ T_s \sin(\theta_0 + T_s \sum_{i=0}^{k-1} \omega_i) \end{bmatrix}}_{B_k} u_k + \boldsymbol{\eta}_k \tag{5.24}$$

with $\boldsymbol{x}_k \in \mathbb{R}^2$ as the two-dimensional position of the car, $u_k \in [0, u_{\max}]$ as the heading velocity, sampling time T_s, known initial heading θ_0, time horizon N, known sequence

of turning rates $\{\omega_k\}_{k=0}^{N-1}$, and a Gaussian random process $\eta_k \sim \mathcal{N}(\overline{\mu}_{\boldsymbol{\eta}}, \Sigma_{\boldsymbol{\eta}})$. The dynamics (5.24) are obtained using the observation that when the turning rate sequence and the initial heading are known, one can *apriori* construct the resulting sequence of heading angles. For a fixed heading velocity, $u_k = v,\ \forall k \in \mathbb{N}_{[0,N]}$, for some $v \in \mathbb{R}$, let $\{\overline{c}_k\}_{k=0}^{N-1}$ be the resulting *nominal* trajectory of (5.24).

We choose the parameters of the problem as $N = 50$, $T_s = 0.1$, $\omega_k = 0.5,\ \forall k \in \mathbb{N}_{[0,N-1]}$, $\theta_0 = \frac{\pi}{4}$, $v = 10$, $u_{\max} = \frac{2v}{3}$, $\overline{\mu}_{\boldsymbol{\eta}} = \overline{0}_2$, and $\Sigma_{\boldsymbol{\eta}} = 10^{-3} I_2$. We are interested in the 0.8-level stochastic reach set of the target tube $\mathcal{T}_k = \text{Box}\left(\overline{c}_k, 0.5 \exp\left(\frac{-k}{N_c}\right)\right),\ \forall k \in \mathbb{N}_{[0,N]}$ where $N_c = 100$ is the decay time constant.

`SReachTools` (via `SReachSet` with option `chance-open`, or equivalently Algorithm 6 using convex convex chance constraints) solves this problem in 137.43 seconds for $|\mathcal{D}| = 16$. In contrast to Section 5.8.2, the conservativeness introduced by Boole's inequality is more severe — the simulated maximal reach probability is 0.15 above the chance constrained estimate. Due to the size of the state space involved, dynamic programming is not feasible. One might be able to use time and state-dependent gridding to compute an approximate solution.

Figure 5.11 shows the mean trajectory for each controller synthesis method available with `SReachPoint`. The computation times, in seconds, were 27.28 for `particle-open`, 527.85 for `genzps-open`, 279.11 for `chance-affine`, and 1.80 for `chance-open`.

5.8.4 Application: Automated Anesthesia Delivery System

We consider the problem of providing probabilistic guarantees of safety for the automated anaesthesia delivery problem [Aba+18]. Uncertainty in the system dynamics is captured via additive Gaussian noise.

The concentration of a drug administered to a patient is often represented by a multi-compartment model [Hil04]. The concentration of Propofol in different com-

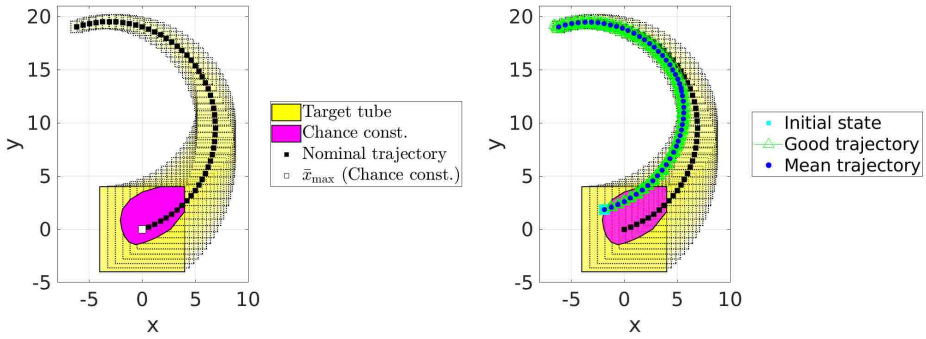


Figure 5.10: Verification and open-loop controller synthesis for a Dubin's vehicle problem with a target tube converging to a nominal trajectory. Monte Carlo simulations (10^5 scenarios with five randomly chosen trajectories displayed) show a simulated reach probability of 0.95; the chance constraint estimate was 0.8.

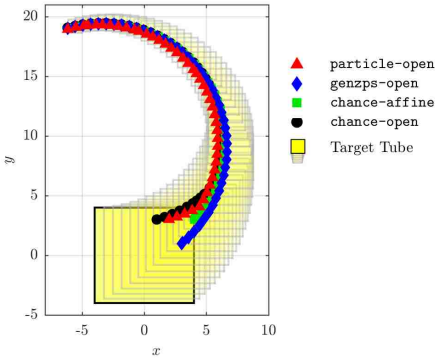


Figure 5.11: Mean trajectories of the Dubins' vehicle corresponding to the optimal controllers synthesized via `SReachPoint`. All trajectories report $r^{\pi^*}_{\bar{x}_0}(\mathscr{T}) \geq 0.9$.

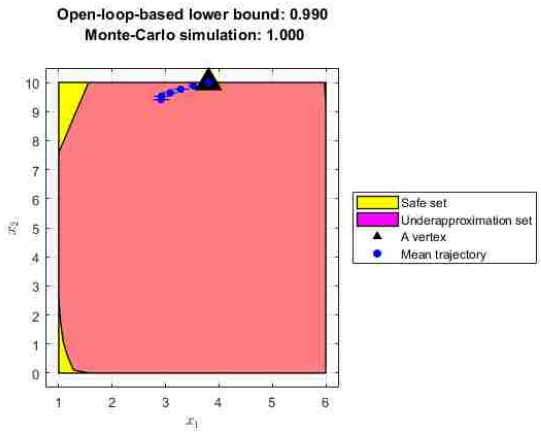


Figure 5.12: Stochastic verification of the automated anesthesia delivery system for $(\boldsymbol{x}_0)_3$ fixed at 5 units. The optimal controller from a vertex was validated using Monte-Carlo simulations (blue ellipses contain 100 randomly choosen trajectories from 10^5 Monte Carlo simulations).

partments of the body are modelled using the three-compartment pharmacokinetic system [GDM14]:

$$\dot{\overline{x}}(t) = \begin{bmatrix} -(k_{10}+k_{12}+k_{13}) & k_{12} & k_{13} \\ k_{21} & -k_{21} & 0 \\ k_{31} & 0 & -k_{31} \end{bmatrix} \overline{x}(t) + \begin{bmatrix} \frac{1}{V_1} \\ 0 \\ 0 \end{bmatrix} u(t) \tag{5.25}$$

The state $\overline{x}(t) \in \mathcal{X} = \mathbb{R}^3$ represents the concentration in each of the three compartments. The input $u(t) \in \mathcal{U} \subset \mathbb{R}$ represents the rate of administration of Propofol, and $V_1, k_{ij} \in \mathbb{R}$ $(i, j \in \{1, 2, 3\})$ are patient dependent parameters. Table 5.4 lists the parameter values determined for a 11-year old child weighing 35 kilograms from the Paedfusor dataset [AK05].

k_{10}	k_{12}	k_{13}	k_{21}	k_{31}	V_1
0.4436	0.1140	0.0419	0.0550	0.0033	16.044

Table 5.4: Model Parameters from the Paedfusor dataset.

We discretize the continuous-time model (5.25) using a zero-order hold with a time-step of $T_s = 20s$. The discrete-time approximation of (5.25) (using parameter values given in Table 5.4) is

$$\boldsymbol{x}_{k+1} = \begin{bmatrix} 0.8192 & 0.03412 & 0.01265 \\ 0.01646 & 0.9822 & 0.0001 \\ 0.0009 & 0.00002 & 0.9989 \end{bmatrix} \boldsymbol{x}_k + \boldsymbol{\omega}_k + \begin{bmatrix} 0.01883 \\ 0.0002 \\ 0.00001 \end{bmatrix} u_k \tag{5.26}$$

$$= A\boldsymbol{x}_k + \boldsymbol{\omega}_k + Bu_k \tag{5.27}$$

The automated input is $u_k \in \mathcal{U}$ and the zero-mean Gaussian random vector $\boldsymbol{\omega}_k \sim \mathcal{N}(0, C)$ with known covariance matrix $C \in \mathbb{R}^{3\times 3}$ accounts for the variation in the system model for different patients. We choose $\mathcal{U} = [0, 7]$ mg/min and $C = 10^{-3} I_3$.

The state $\boldsymbol{x}_k$ is a random vector, due to the stochastic noise $\boldsymbol{\omega}_k$.

Consider a safe set $\mathcal{K} = \{\overline{z} \in \mathcal{X} : 1 \leq \overline{z}_1 \leq 6, 0 \leq \overline{z}_2 \leq 10, 0 \leq \overline{z}_3 \leq 10\} \subset \mathcal{X}$, and a target set $\mathcal{T} = \{\overline{z} \in \mathcal{X} : 4 \leq \overline{z}_1 \leq 6, 8 \leq \overline{z}_2 \leq 10, 8 \leq \overline{z}_3 \leq 10\} \subset \mathcal{X}$. We wish to verify (5.26) with respect to the safe set $\mathcal{K}$ for a time horizon of 5 time steps.

Figure 5.12 shows the stochastic verification of the automated anesthesia delivery system. The verification was done with a safety probability threshold of 0.99 using `SReachTools` (via `SReachSet` with option `chance-open`, or equivalently Algorithm 6 using convex convex chance constraints). The computation took 16.57 seconds to analyze 64 directions, sampled from a unit circle such that they subtend the same angle at the center.

5.9 Summary

This chapter discusses four novel approaches for controller synthesis for the stochastic reachability of a target tube. We build our grid-free and scalable approaches using the set-theoretical results presented in Chapter 4. Specifically, we demonstrate underapproximative and approximative point-based stochastic reachability evaluations along with open-loop or affine feedback controller synthesis using convex optimization, stochastic programming, and Fourier transforms. We also combine these techniques with a ray-shooting algorithm to underapproximate the sets in a scalable, grid-free, and anytime algorithm. This approach enables, for the first time, the verification of systems as high as 40-dimensions.

Chapter 6

SReachTools: A MATLAB Toolbox for Stochastic Reachability

6.1 Introduction

This chapter briefly discusses the features of `SReachTools`, an open-source MATLAB toolbox that implements all the algorithms discussed in Chapter 5. The primary goal of `SReachTools` is to allow an end-user to solve stochastic reachability problems without requiring exhaustive knowledge of the underlying algorithms or their implementation.

The toolbox is available online at https://unm-hscl.github.io/SReachTools/. The project website also has a comprehensive documentation of the various functions in `SReachTools` as well as demonstration examples.

6.2 Features of SReachTools

Table 6.1 shows the primary solving functions in `SReachTools` and the different solution methods available for each function along with their utility.

Function	`method_str`	Utility
`SReachDyn`	–	Dynamic programming-based approximation of $r_{\overline{x}_0}^{\pi^*}(\mathscr{T})$ and $\mathcal{L}_0^{\pi^*}(\alpha, \mathscr{T})$
`SReachPoint`	`chance-open`	Underapproximation of $r_{\overline{x}_0}^{\pi^*}(\mathscr{T})$; Implements (5.15)
	`genzps-open`	Approximation of $r_{\overline{x}_0}^{\pi^*}(\mathscr{T})$ within ϵ_{genz}; Implements (5.14) using Genz's algorithm [Gen14]
	`particle-open`	Approximation of $r_{\overline{x}_0}^{\pi^*}(\mathscr{T})$; Implements (5.17) [LOE13]
	`chance-affine`	Underapproximation of $r_{\overline{x}_0}^{\overline{M}^*,\overline{D}^*}(\mathscr{T}; \Delta_U)$; Implements (5.8)
`SReachSet`	`chance-open`	Implements Alg. 6 using convex chance constraints to underapproximate $\mathcal{L}_0^{\pi^*}(\alpha, \mathscr{T})$
	`genzps-open`	Implements Alg. 6 using Genz's algorithm [Gen14] to underapproximate $\mathcal{L}_0^{\pi^*}(\alpha, \mathscr{T})$
	`lag-open`	Lagrangian approach [GVO17]
	`lag-under`	
`SReachFwd`	`state-stoch`	Stochasticity of $\boldsymbol{x}_k$
	`concat-stoch`	Stochasticity of $\boldsymbol{X}$
	`state-prob`	$\mathbb{P}_{\boldsymbol{x}_k}^{x_0}\{x_k \in \mathcal{T}\}$
	`concat-prob`	$\mathbb{P}_{\boldsymbol{X}}^{x_0}\{\forall i \in \mathbb{N}_{[1,k]},\ x_i \in \mathcal{T}_i\}$

Table 6.1: Current features of `SReachTools`. Here, $r_{\overline{x}_0}^{\pi^*}(\mathscr{T})$ corresponds to the maximum reach probability for staying within the target tube $\mathscr{T}$ (4.2), and $\mathcal{L}_0^{\pi^*}(\alpha, \mathscr{T})$ is the set of initial states from which the maximal reach probability is above a user-specified threshold.

6.2.1 Auxillary Functions

`SReachTools` comes with a built-in initialization script `srtinit` and a broad set of unit-testing functions to ensure proper performance of the toolbox. The initialization script is designed to eliminate potential function-name overloading by only adding the `SReachTools` source functions to the MATLAB path when they are in use.

For easy handling of linear, discrete-time systems, we have developed `LtiSystem` and `LtvSystem` classes to define linear systems (2.19). Both of these classes also permit specification of the additive disturbance as $\boldsymbol{w}_k = F_k \boldsymbol{v}_k$ as well, for some random vector $\boldsymbol{v}_k$. We use the `Multi-Parameteric Toolbox (MPT3)` [Her+13] to define polytopic input spaces and polytopic disturbance spaces, if required. `SReachTools` has several demonstration systems, such as a chain of integrators, Dubins car, and Clohessy-Wiltshire-Hill spacecraft near-orbit relative dynamics.

`SReachTools` also has the `Tube` class to define polytopic target tubes, and `RandomVector` class to define random vector disturbances or initial states. `Tube` is implemented using `MPT3` [Her+13] to define a collection of time-stamped polytopic safe sets. One can also easily specify a reach-avoid [SL10] or viability [Aba+08] specification using `Tube`. Currently, `RandomVector` supports only Gaussian disturbances, while development is underway to include non-Gaussian disturbances.

Using objects of `Tube`, `RandomVector`, and `LtiSystem` or `LtvSystem` classes, one can pose the problems of stochastic reachability discussed in Section 4.3. We presume for the remainder that `sys` refers to an instance of the `LtiSystem` or `LtvSystem` classes to describe (2.19) respectively, and `tube` refers to an instance of the `Tube` class to describe $\mathscr{T}$.

6.2.2 Stochastic Reachability of a Target Tube

Dynamic Programming

We can use `SReachTools` to implement a grid-based implementation of (4.3) using `SReachDynProg`.

```
% Dynamic programming solution with SReachTools
SReachDynProg(prob_str, sys, x_inc, u_inc, tube)
```

Here, `prob_str` refers to the type of reachability problem, `x_inc` and `u_inc` are grid step sizes for the state space $\mathcal{X}$ and input space $\mathcal{U}$ respectively. Currently in the `SReachTools` toolbox, `prob_str` must always be 'term', as version 1.0.0 is only designed for the problem of stochastic reachability of a target tube, i.e. the terminal-time hitting problem. Future releases will address other reachability problems, e.g. the first-time hitting problem [SL10].

Point-Based Stochastic Reachability Computation

`SReachTools` implements all the algorithms proposed in Chapter 5. Using `SReachPoint`, it can compute an (open-loop or affine) optimal controller from a given initial state that maximizes the probability of the state of the stochastic linear system (2.19) to stay within a pre-specified target tube $\mathscr{T}$. This function implements the discussion in Sections 5.5, 5.6.1, 5.6.2, and 5.6.3. We use `CVX` [GB14] to implement all of the convex and mixed-integer linear programs.

```
% Generate SReachTools options
options = SReachPointOptions(prob_str, method_str)
% Point-based stochastic target tube reachability
SReachPoint(prob_str, method_str, sys, init_state, tube, options)
```

Here, `method_str` refers to the optimization approach ('`chance-open`', '`particle-open`', '`genzps-open`', or '`chance-affine`'), `init_state` is the initial state $x_0 \in \mathcal{X}$ at which an (under-)approximation of $r^{\pi^*}_{x_0}(\mathscr{T})$ must be evaluated, and

`options` enables specification of accuracy and method-specific arguments (i.e., Δ_U).

Stochastic Reach Set Computation

`SReachTools` implements the polytopic approximation of the stochastic reach set using `SReachSet`. It implements Algorithm 6 in Chapter 5 using the open-loop controller-based reach probability using Genz's algorithm or chance constraints (see Table 5.1). In addition, `SReachSet` also implements the Lagrangian technique to compute (over- and under-) approximations to the stochastic reach set $\mathcal{L}_0^{\pi^*}(\alpha, \mathscr{T})$. The Lagrangian approach uses a backward recursion that employs Minkowski sum, difference, intersection, and affine transformations [GVO17].

```
% Generate SReachTools options
options = SReachSetOptions(prob_str, method_str)
% Compute stochastic reach sets with SReachTools
SReachSet(prob_str, method_str, sys, thresh, tube, options)
```

Here, `method_str` can be `'lag-under'` or `'lag-over'` for the Lagrangian under- and over-approximation, respectively; `'genz-open'` for an under-approximation using Genz algorithm; and `'chance-open'` for an under-approximation using chance-constrained optimization. The input `'thresh'` is the probabilistic bound, α, and `options` is used to specify additional solver options like direction vectors to use for the point-based stochastic reach set computations.

6.2.3 Forward Stochastic Reachability

`SReachTools` characterizes the mean and covariance of x_k and $\overline{x}_{[0,k]}$ using the fact that affine transformations of Gaussian random vectors are Gaussian [Gub06, Sec. 9.2] (i.e., the prediction step of a Kalman filter). We employ Genz's algorithm [Gen92; Gen14] for numerical evaluation of the high-dimensional quadrature of multivariate Gaussians over polytopes, up to a user-specified accuracy of ϵ_{genz}.

```
% Forward stochastic reachability via SReachTools
SReachFwd(prob_str, sys, init_state, k, [set/tube])
```

Table 6.1 describes various values `prob_str` can take. The input `init_state` takes in the initial state, which can be a deterministic vector or a random vector (an object of `RandomVector`). `SReachFwd` needs a `set` (target set $\mathcal{T}$) or `tube` (target tube $\mathscr{T}$), only if `prob_str` is '`state-prob`' or '`concat-prob`', respectively.

Chapter 7

Conclusion

This dissertation proposes novel theory and scalable algorithms for tractable solution to the stochastic reachability problems. Our motivation stems from the need for safety and performance guarantees in control systems. These guarantees for complex control systems with stochastic uncertainty allow for insightful analysis, especially when failure to meet a given objective, while possible, is unlikely. We demonstrate the utility of this work in a host of applications: stochastic target capture (Section 3.6), stochastic motion planning (Section 3.7), verification of spacecraft rendezvous problem (Section 5.8.2), and automated anesthesia delivery system (Section 5.8.4).

7.1 Summary of Contributions

We first analyzed the problem of forward stochastic reachability, the problem of predicting the stochasticity of the state at a future time instant. Using Fourier transforms and computational geometry, we discussed a grid-free, recursion-free, and scalable approach to estimate the associated probability density. We also characterized the sufficient conditions under which this density is log-concave and the reach set is convex. These convexity results enabled a convex optimization-based formulation of the stochastic target capture problem, enabling computationally efficient search for glob-

ally optimal solutions. We experimentally validated this approach on a quadrotor testbed.

For the problem of obstacle prediction in stochastic motion planning problems, we used forward stochastic reachability analysis to define probabilistic occupancy function and α-probabilistic occupied sets. We characterized multiple grid-free and scalable algorithms to compute polytopic and ellipsoidal overapproximations of these sets. These algorithms relied on the proposed sufficient conditions for the convexity and compactness of α-probabilistic occupied set. These sets can be used with any standard motion planning algorithms capable of avoiding dynamic obstacles to generate probabilistically safe trajectories. We demonstrated the utility of these algorithms by using successive convexification techniques to solve the stochastic motion planning problem for linear robot dynamics and multiple rigid-body obstacles with stochastic linear dynamics.

Next, we analyzed the problem of backward stochastic reachability by formulating the problem of stochastic reachabilty of a target tube. This problem subsumed the existing work on terminal hitting-time stochastic reach-avoid problems and stochastic viability problems. Of special interest is the stochastic reach set, the set of all initial states that can be driven to a probabilistic safety above a threshold under bounded control authority and system dynamics. We characterized sufficient conditions under which this set exists, and is closed, bounded, convex, and compact. The convexity and compactness results enabled efficient computation of underapproximative polytopic underaproximations of these sets. We also synthesized open-loop and affine feedback controllers to maximize the safety probability, and discussed an underapproximative interpolation technique for stochastic reach sets. We demonstrated the utility of this approach on various applications including verification of spacecraft rendezvous problem, automated anesthesia delivery, and stochastic motion planning.

Finally, we presented an open-source MATLAB toolbox, `SReachTools`, that im-

plements all the algorithms presented in this dissertation in an easily-accessible code base.

7.2 Future Directions

This dissertation opens several new doors with its theoretical analysis and scalable algorithms. We describe few of the exciting directions that may be explored next.

- *Stochastic reachability for analysis of human-in-the-loop systems*: Several modern-day control systems involve a human in the control loop. The scalable techniques for verification proposed in this dissertation can enable accounting for the human as a stochastic agent affecting the system, as opposed to an adversarial agent, as is typically done in literature [GDCB17].

- *Extensions of the computational approaches to systems beyond linear systems* (2.19): We required linearity of the system to guarantee convexity (see Table 4.1). Therefore, a generic nonlinear equality constraint will render the optimization problem non-convex. On the other hand, semidefinite programs have been used to analyze stochastic nonlinear systems [Hor14].

- *Theory and algorithms for the first hitting-time stochastic reach-avoid problems*: Informally, the first hitting-time stochastic reach-avoid problems optimizes for an admissible controller that maximizes the probability of reaching a target set *within* the time horizon, while staying in a safety tube [SL10]. In contrast, the problem of stochastic reachability of a target tube requires the target set be hit precisely at the time horizon. First hitting-time stochastic reach-avoid problems are relevant in problems of stochastic motion planning, where a better maneuver might be missed if a target tube formulation is used. Chapters 4 and 5 provided sufficient conditions for set properties like convexity and compactness

of the stochastic reach set for the terminal hitting-time problem. We should explore if these properties and algorithms can be extended to the first hitting-time problem. Currently, approximate dynamic programming approaches are the only available approach for this problem [Kar+17], and they lack guarantees in the direction of approximation.

References

[Aba+07] A. Abate, S. Amin, M. Prandini, J. Lygeros, and S. Sastry. "Computational approaches to reachability analysis of stochastic hybrid systems". In: *Proc. Hybrid Syst.: Comput. & Ctrl.* 2007, pp. 4–17.

[Aba+08] A. Abate, M. Prandini, J. Lygeros, and S. Sastry. "Probabilistic reachability and safety for controlled discrete time stochastic hybrid systems". In: *Automatica* 44.11 (2008), pp. 2724–2734.

[Aba+18] A. Abate, H. Blom, N. Cauchi, S. Haesaert, A. Hartmanns, K. Lesser, M. Oishi, V. Sivaramakrishnan, S. Soudjani, C. Vasile, et al. "ARCH-COMP18 Category Report: Stochastic Modelling". In: *Proc. Int'l Workshop on App. Verification Cont. and Hybrid Syst. (ARCH)*. Vol. 54. 2018, pp. 71–103.

[AS65] M. Abramowitz and I. Stegun. *Handbook of Mathematical Functions: with Formulas, Graphs, & Mathematical Tables*. Courier Corp., 1965.

[AK05] A. Absalom and G. Kenny. "'Paedfusor' pharmacokinetic data set". In: *British Journal of Anaesthesia* 95.1 (2005), pp. 110–110.

[ASB09] M. Althoff, O. Stursberg, and M. Buss. "Model-based probabilistic collision detection in autonomous driving". In: *IEEE Trans. on Intel. Transp. Syst.* 10.2 (2009), pp. 299–310.

[Aou+13] G. Aoude, B. Luders, J. Joseph, N. Roy, and J. How. "Probabilistically safe motion planning to avoid dynamic obstacles with uncertain motion patterns". In: *Autonomous Robots* 35.1 (2013), pp. 51–76.

[BMGF10] A. Bautin, L. Martinez-Gomez, and T. Fraichard. "Inevitable collision states: a probabilistic perspective". In: *Proc. IEEE Int'l Conf. Robotics & Autom.* IEEE. 2010, pp. 4022–4027.

[BR71] D. Bertsekas and I. Rhodes. "On the minimax reachability of target sets and target tubes". In: *Automatica* 7.2 (1971), pp. 233–247.

[BS78] D. Bertsekas and S. Shreve. *Stochastic optimal control: The discrete time case*. Academic Press, 1978.

[Ber72] D. Bertsekas. "Infinite time reachability of state-space regions by using feedback control". In: *IEEE Trans. Autom. Ctrl.* 17.5 (1972), pp. 604–613.

[Bil95] P. Billingsley. *Probability and measure*. 3rd ed. Wiley, 1995.

[BHW06] L. Blackmore, Hui Li, and B. Williams. "A probabilistic approach to optimal robust path planning with obstacles". In: *Proc. Amer. Ctrl. Conf.* 2006, pp. 2831–2837.

[BOW11] L. Blackmore, M. Ono, and B. Williams. "Chance-constrained optimal path planning with obstacles". In: *IEEE Trans. Robot.* 27.6 (2011), pp. 1080–1094.

[BFZ10] O. Bokanowski, N. Forcadel, and H. Zidani. "Reachability and Minimal Times for State Constrained Nonlinear Problems without Any Controllability Assumption". In: *SIAM J. of Ctrl. & Opt.* 48.7 (2010), pp. 4292–4316.

[BV04] S. Boyd and L. Vandenberghe. *Convex optimization*. Cambridge Univ. Press, 2004.

[Bra86] R. Bracewell. *The Fourier transform and its applications*. McGraw-Hill, Inc., 1986.

[CC06] G. Calafiore and M. Campi. "The scenario approach to robust control design". In: *IEEE Trans. Autom. Cntrl* 51.5 (2006), pp. 742–753.

[Chi+15] H.-T. Chiang, N. Malone, K. Lesser, M. Oishi, and L. Tapia. "Aggressive moving obstacle avoidance using a stochastic reachable set based potential field". In: *Algo. Found. of Robotics*. Springer, 2015, pp. 73–89.

[Chi+17] H.-T. L. Chiang, B. HomChaudhuri, A. P. Vinod, M. Oishi, and L. Tapia. "Dynamic risk tolerance: Motion planning by balancing short-term and long-term stochastic dynamic predictions". In: *Robotics and Automation (ICRA), 2017 IEEE International Conference on*. IEEE. 2017, pp. 3762–3769.

[CT97] Y. Chow and H. Teicher. *Probability Theory: Independence, Interchangeability, Martingales*. 3rd ed. Springer New York, 1997.

[DJD88] S. Dharmadhikari and K. Joag-Dev. *Unimodality, convexity, and applications*. Elsevier, 1988.

[Din+13] J. Ding, M. Kamgarpour, S. Summers, A. Abate, J. Lygeros, and C. Tomlin. "A stochastic games framework for verification and control of discrete time stochastic hybrid systems". In: *Automatica* 49.9 (2013), pp. 2665–2674.

[DCA94] P. Dorato, V. Cerone, and C. Abdallah. *Linear-quadratic control: An introduction*. Simon & Schuster, 1994.

[Drz+16] D. Drzajic, N. Kariotoglou, M. Kamgarpour, and J. Lygeros. "A Semidefinite Programming Approach to Control Synthesis for Stochastic Reach-Avoid Problems". In: *Proc. Int'l Workshop on App. Verification Cont. and Hybrid Syst. (ARCH)*. 2016, pp. 134–143.

[DTB11] N. Du Toit and J. Burdick. "Probabilistic collision checking with chance constraints". In: *IEEE Trans. Robot.* 27.4 (2011), pp. 809–815.

[Elf89] A. Elfes. "Using occupancy grids for mobile robot perception and navigation". In: *Computer* 22.6 (1989), pp. 46–57.

[FGS16] M. Farina, L. Giulioni, and R. Scattolini. "Stochastic linear model predictive control with chance constraints–a review". In: *J. Process Ctrl.* 44 (2016), pp. 53–67.

[FSL07] C. Fulgenzi, A. Spalanzani, and C. Laugier. "Dynamic Obstacle Avoidance in uncertain environment combining PVOs and Occupancy Grid". In: *Proc. IEEE Int'l Conf. Robotics & Autom.* Apr. 2007, pp. 1610–1616.

[GK] P. Gagarinov and A. Kurzhanskiy. *Ellipsoidal Toolbox*.

[GDM14] V. Gan, G. A. Dumont, and I. Mitchell. "Benchmark Problem: A PK/PD Model and Safety Constraints for Anesthesia Delivery". In: *ARCH@ CPSWeek*. 2014, pp. 1–8.

[Gen92] A. Genz. "Numerical computation of multivariate normal probabilities". In: *J. of Comp. and Graph. Stat.* 1.2 (1992), pp. 141–149.

[Gen14] A. Genz. *Quadrature of a multivariate normal distribution over a region specified by linear inequalities: QSCMVNV*. http://www.math.wsu.edu/faculty/genz/software/matlab/qscmvnv.m. 2014.

[Gey08] C. Geyer. "Active target search from UAVs in urban environments". In: *Proc. IEEE Int'l Conf. Robotics & Autom.* 2008, pp. 2366–2371.

[Gir05] A. Girard. "Reachability of uncertain linear systems using zonotopes". In: *Proc. Hybrid Syst.: Comput. & Ctrl.* 2005, pp. 291–305.

[GVO17] J. Gleason, A. Vinod, and M. Oishi. "Underapproximation of Reach-Avoid Sets for Discrete-Time Stochastic Systems via Lagrangian Methods". In: *Proc. IEEE Conf. Dec. & Ctrl.* 2017.

[Goo18] Google. *Waymo: Google's self-driving car project*. 2018. URL: https://waymo.com/.

[GKM06] P. Goulart, E. Kerrigan, and J. Maciejowski. "Optimization over state feedback policies for robust control with constraints". In: *Automatica* 42 (2006), pp. 523–533.

[GDCB17] V. Govindarajan, K. Driggs-Campbell, and R. Bajcsy. "Data-driven reachability analysis for human-in-the-loop systems". In: *Decision and Control (CDC), 2017 IEEE 56th Annual Conference on*. IEEE. 2017, pp. 2617–2622.

[GB14] M. Grant and S. Boyd. *CVX: MATLAB Software for Disciplined Convex Programming, version 2.1*. http://cvxr.com/cvx. 2014.

[Gub06] J. Gubner. *Probability and random processes for electrical and computer engineers*. Cambridge Univ. Press, 2006.

[Gur15] Gurobi Optimization Inc. *Gurobi Optimizer Reference Manual*. 2015. URL: http://www.gurobi.com.

[HL10] R. Harman and V. Lacko. "On decompositional algorithms for uniform sampling from n-spheres and n-balls". In: *J. of Multi. Anal.* 101.10 (2010), pp. 2297–2304.

[Har59] P. Hartman. "On functions representable as a difference of convex functions". In: *Pacific J. Mathematics* 9.3 (1959), pp. 707–713.

[Her+13] M. Herceg, M. Kvasnica, C. Jones, and M. Morari. "Multi-Parametric Toolbox 3.0". In: *Proc. Euro. Ctrl. Conf.* http://mpt3.org. 2013, pp. 502–510.

[Hil04] S. Hill. "Pharmacokinetics of drug infusions". In: *Continuing education in anesthesia, critical care & Pain* 4.3 (2004), pp. 76–80.

[HPV76] C. Himmelberg, T. Parthasarathy, and F. VanVleck. "Optimal Plans for Dynamic Programming Problems". In: *Mathematics of Operations Research* 1.4 (1976), pp. 390–394.

[Hok+12] P. Hokayem, E. Cinquemani, D. Chatterjee, F. Ramponi, and J. Lygeros. "Stochastic receding horizon control with output feedback and bounded controls". In: *Automatica* 48 (2012), pp. 77–88.

[Hol+09] G. Hollinger, S. Singh, J. Djugash, and A. Kehagias. "Efficient multi-robot search for a moving target". In: *Int'l J. Robotics & Res.* 28.2 (2009), pp. 201–219.

[HVO17] B. HomChaudhuri, A. Vinod, and M. Oishi. "Computation of forward stochastic reach sets: Application to stochastic, dynamic obstacle avoidance". In: *Proc. Amer. Ctrl. Conf.* 2017, pp. 4404–4411.

[Hor14] M. Horowitz. "Efficient methods for stochastic optimal control". PhD thesis. California Institute of Technology, 2014. URL: http://thesis.library.caltech.edu/8453/.

[HPT00] R. Horst, P. Pardalos, and N. Thoai. *Introduction to global optimization*. Springer Science & Business Media, 2000.

[Hua+15] H. Huang, J. Ding, W. Zhang, and C. Tomlin. "Automation-Assisted Capture-the-Flag: A Differential Game Approach". In: *IEEE Trans. Ctrl. Syst. Tech.* 23 (2015), pp. 1014–1028.

[Ich+17] B. Ichter, E. Schmerling, A. Agha-mohammadi, and M. Pavone. "Real-time stochastic kinodynamic motion planning via multiobjective search on GPUs". In: *Proc. IEEE Int'l Conf. Robotics & Autom.* 2017, pp. 5019–5026.

[KF11] S. Karaman and E. Frazzoli. "Sampling-based algorithms for optimal motion planning". In: *Int'l J. Robotics & Res.* 30.7 (2011), pp. 846–894.

[KML16] N. Kariotoglou, K. Margellos, and J. Lygeros. "On the computational complexity and generalization properties of multi-stage and stage-wise coupled scenario programs". In: *Syst. & Ctrl. Lett.* 94 (2016), pp. 63–69.

[Kar+11] N. Kariotoglou, D. M. Raimondo, S. Summers, and J. Lygeros. "A stochastic reachability framework for autonomous surveillance with pan-tilt-zoom cameras". In: *Proc. Euro. Ctrl. Conf.* 2011, pp. 1411–1416.

[Kar+14] N. Kariotoglou, D. Raimondo, S. Summers, and J. Lygeros. "Multi-agent autonomous surveillance: a framework based on stochastic reachability and hierarchical task allocation". In: *J. Dyn. Sys., Meas., Ctrl.* 137.3 (2014).

[Kar+17] N. Kariotoglou, M. Kamgarpour, T. H. Summers, and J. Lygeros. "The linear programming approach to reach-avoid problems for Markov decision processes". In: *Journal of Artificial Intelligence Research* 60 (2017), pp. 263–285.

[KLT03] T. Kolda, R. Lewis, and V. Torczon. "Optimization by direct search: New perspectives on some classical and modern methods". In: *SIAM review* 45.3 (2003), pp. 385–482.

[KRS04] V. Kumar, D. Rus, and S. Singh. "Robot and sensor networks for first responders". In: *IEEE Pervasive computing* 3.4 (2004), pp. 24–33.

[KV97] A. Kurzhanski and I. Vályi. *Ellipsoidal calculus for estimation and control*. Laxenburg, Austria, 1997.

[KV00] A. Kurzhanski and P. Varaiya. "Ellipsoidal Techniques for Reachability Analysis". In: *Proc. Hybrid Syst.: Comput. & Ctrl.* (2000), pp. 202–214.

[KV06] A. Kurzhanskiy and P. Varaiya. *Ellipsoidal toolbox*. Tech. rep. UCB/EECS-2006-46. EECS Department, University of California, Berkeley, 2006.

[Kva+15] M. Kvasnica, B. Takács, J. Holaza, and D. Ingole. "Reachability Analysis and Control Synthesis for Uncertain Linear Systems in MPT". In: *IFAC Symp. on Robust Control D.* 48.14 (2015), pp. 302–307.

[LGSP08] A. Lambert, D. Gruyer, and G. Saint Pierre. "A fast monte carlo algorithm for collision probability estimation". In: *International Conference on Autom., Rob., Ctrl., and Vis.* IEEE. 2008, pp. 406–411.

[LaV06] S. LaValle. *Planning Algorithms*. Cambridge Univ. Press, 2006.

[LGG09] C. Le Guernic and A. Girard. "Reachability analysis of hybrid systems using support functions". In: *Int'l Conf. on Comput. Aided Verification*. Springer, 2009, pp. 540–554.

[LOE13] K. Lesser, M. Oishi, and R. Erwin. "Stochastic reachability for control of spacecraft relative motion". In: *Proc. IEEE Conf. Dec. & Ctrl.* 2013.

[LB16] T. Lipp and S. Boyd. "Variations and extension of the convex–concave procedure". In: *Optim. and Engg.* 17 (2016), pp. 263–287.

[LKH10] B. Luders, M. Kothari, and J. How. "Chance Constrained RRT for Probabilistic Robustness to Environmental Uncertainty". In: *J. Guid., Navig., and Cntrl.* AIAA, Aug. 2010.

[Mal+17] N. Malone, H.-T. Chiang, K. Lesser, M. Oishi, and L. Tapia. "Hybrid Dynamic Moving Obstacle Avoidance Using a Stochastic Reachable Set-Based Potential Field". In: *IEEE Trans. Robot.* 33.5 (2017), pp. 1124–1138.

[Man+15] G. Manganini, M. Pirotta, M. Restelli, L. Piroddi, and M. Prandini. "Policy search for the optimal control of Markov Decision Processes: A Novel Particle-Based Iterative Scheme". In: *IEEE Trans. Cybern.* (2015).

[MSA16] Y. Mao, M. Szmuk, and B. Açıkmeşe. "Successive convexification of non-convex optimal control problems and its convergence properties". In: *Proc. IEEE Conf. Dec. & Ctrl.* 2016, pp. 3636–3641.

[Mao+17] Y. Mao, D. Dueri, M. Szmuk, and B. Açıkmeşe. "Successive Convexification of Non-Convex Optimal Control Problems with State Constraints". In: *IFAC-PapersOnLine* 50.1 (2017), pp. 4063 –4069.

[MW08] Masahiro Ono and B. Williams. "Iterative risk allocation: A new approach to robust model predictive control with a joint chance constraint". In: *Proc. IEEE Conf. Dec. & Ctrl.* 2008, pp. 3427–3432.

[MKK12] D. Mellinger, A. Kushleyev, and V. Kumar. "Mixed-integer quadratic program trajectory generation for heterogeneous quadrotor teams". In: *Proc. IEEE Int'l Conf. Robotics & Autom.* 2012, pp. 477–483.

[Mes16] A. Mesbah. "Stochastic model predictive control: An overview and perspectives for future research". In: *IEEE Ctrl. Syst. Mag.* 36.6 (2016), pp. 30–44.

[MBT05] I. Mitchell, A. Bayen, and C. Tomlin. "A time-dependent Hamilton-Jacobi formulation of reachable sets for continuous dynamic games". In: *IEEE Trans. Autom. Ctrl.* 50.7 (2005), pp. 947–957.

[MT00] I. Mitchell and C. Tomlin. "Level set methods for computation in hybrid systems". In: *Proc. Hybrid Syst.: Comput. & Ctrl.* 2000, pp. 310–323.

[Old+14] F. Oldewurtel, C. Jones, A. Parisio, and M. Morari. "Stochastic model predictive control for building climate control". In: *IEEE Trans. Ctrl. Syst. Tech.* 22.3 (2014), pp. 1198–1205.

[OW08] M. Ono and B. Williams. “Iterative risk allocation: a new approach to robust Model Predictive Control with a joint chance constraint”. In: *Proc. IEEE Conf. Dec. & Ctrl.* 2008, pp. 3427–3432.

[Ono+15] M. Ono, M. Pavone, Y. Kuwata, and J Balaram. “Chance-constrained dynamic programming with application to risk-aware robotic space exploration”. In: *Autonomous Robots* 39.4 (2015), pp. 555–571.

[Pre80] A. Prekopa. “Logarithmic concave measures and related topics”. In: *Stochastic Programming* (1980).

[Pre+07] W. Press, S. Teukolsky, W. Vetterling, and B. Flannery. *Numerical recipes: The art of scientific computing*. 3rd ed. New York, NY, USA: Cambridge University Press, 2007.

[Put05] M. Putterman. *Markov Decision Processes: Discrete Stochastic Dynamic Programming*. John Wiley & Sons, 2005.

[RW09] R. Rockafellar and R. Wets. *Variational analysis*. Vol. 317. Springer Science & Business Media, 2009.

[Rud87] W. Rudin. *Real and complex analysis*. Tata McGraw-Hill Ed., 1987.

[Sar+19] H. Sartipizadeh, A. Vinod, B. Acikmese, and M. Oishi. “Voronoi partition-based scenario reduction for fast sampling-based stochastic reachability computation of LTI systems”. In: *Proc. Amer. Ctrl. Conf.* (submitted). 2019.

[SFH02] T. Schouwenaars, É. Féron, and J. How. “Safe receding horizon path planning for autonomous vehicles”. In: *Allerton Conf. on Comm., Ctrl., and Comp.* Vol. 40. 2002, pp. 295–304.

[SB10] J. Skaf and S. Boyd. “Design of affine controllers via convex optimization”. In: *IEEE Trans. Autom. Ctrl.* 55.11 (2010), pp. 2476–87.

[SS03] E. Stein and R. Shakarchi. *Fourier analysis: An introduction*. Princeton Univ. Press, 2003.

[SW71] E. Stein and G. Weiss. *Introduction to Fourier analysis on Euclidean spaces*. Vol. 1. Princeton Univ. Press, 1971.

[Str88] G. Strang. *Linear Algebra and its Applications*. 3rd ed. Thomson Learning, Inc, 1988.

[SL10] S. Summers and J. Lygeros. "Verification of discrete time stochastic hybrid systems: A stochastic reach-avoid decision problem". In: *Automatica* 46.12 (2010), pp. 1951–1961.

[Sum+11] S. Summers, M. Kamgarpour, J. Lygeros, and C. Tomlin. "A Stochastic Reach-avoid Problem with Random Obstacles". In: *Proc. Hybrid Syst.: Comput. & Ctrl.* 2011, pp. 251 –260.

[SB98] R. Sutton and A. Barto. *Reinforcement learning - an introduction*. MIT Press, 1998.

[Tao06a] T. Tao. *Analysis*. 2nd ed. Vol. 1. Hindustan Book Agency, 2006.

[Tao06b] T. Tao. *Analysis*. 2nd ed. Vol. 2. Hindustan Book Agency, 2006.

[TBF05] S. Thrun, W. Burgard, and D. Fox. *Probabilistic robotics*. MIT Press, 2005.

[TLS00] C. Tomlin, J. Lygeros, and S. Sastry. "A game theoretic approach to controller design for hybrid systems". In: *Proc. IEEE* 88.7 (2000), pp. 949–970.

[Tom+03] C. Tomlin, I. Mitchell, A. Bayen, and M. Oishi. "Computational techniques for the verification of hybrid systems". In: *Proc. IEEE* 91.7 (2003), pp. 986–1001.

[VHO17] A. Vinod, B. HomChaudhuri, and M. Oishi. "Forward Stochastic Reachability Analysis for Uncontrolled Linear Systems Using Fourier Transforms". In: *Proc. Hybrid Syst.: Comput. & Ctrl.* 2017, pp. 35–44.

[VO17] A. Vinod and M. Oishi. "Scalable Underapproximation for the Stochastic Reach-Avoid Problem for High-Dimensional LTI Systems Using Fourier Transforms". In: *IEEE Letters-Cntrl. Syst. Society* 1.2 (2017), pp. 316–321.

[VO18a] A. Vinod and M. Oishi. "Probabilistic Occupancy Function and Sets Using Forward Stochastic Reachability for Rigid-Body Dynamic Obstacles". In: *IEEE Trans. Autom. Cntrl* (2018). (submitted) https://arxiv.org/abs/1803.07180.

[VO18b] A. Vinod and M. Oishi. "Scalable Underapproximative Verification of Stochastic LTI Systems Using Convexity and Compactness". In: *Proc. Hybrid Syst.: Comput. & Ctrl.* Porto, Portugal, 2018, pp. 1–10.

[VO18c] A. Vinod and M. Oishi. "Stochastic reachability of a target tube: Theory and Computation". In: *IEEE Trans. Autom. Cntrl* (2018). (submitted) https://arxiv.org/abs/1810.05217.

[VO19] A. Vinod and M. Oishi. "Affine controller synthesis for stochastic reachability via difference of convex programming". In: *Proc. Hybrid Syst.: Comput. & Ctrl.* (submitted). 2019.

[VSO19] A. Vinod, V. Sivaramakrishnan, and M. Oishi. "Piecewise-Affine Approximation-Based Stochastic Optimal Control with Gaussian Joint Chance Constraints". In: *Proc. Amer. Ctrl. Conf.* (submitted). 2019.

[Vin+18a] A. Vinod, S. Rice, Y. Mao, M. Oishi, and B. Acikmese. "Stochastic motion planning using successive convexification and probabilistic occupancy functions". In: *Proc. IEEE Conf. Dec. & Ctrl.* (accepted). 2018.

[Vin+18b] A. Vinod, B. HomChaudhuri, C. Hintz, A. Parikh, S. Buerger, M. Oishi, G. Brunson, S. Ahmad, and R. Fierro. "Multiple Pursuer-Based Intercept via Forward Stochastic Reachability". In: *Proceedings of the American Control Conference (ACC)*. 2018, pp. 1559–1566.

[VT11] M. Vitus and C. Tomlin. "On feedback design and risk allocation in chance constrained control". In: *Proc. IEEE Conf. Dec. & Ctrl.* 2011, pp. 734–739.

[Web94] R. Webster. *Convexity*. Oxford University Press, 1994.

[Wie89] W. Wiesel. *Spaceflight Dynamics*. New York: McGraw-Hill, 1989.

[WH12] A. Wu and J. How. "Guaranteed infinite horizon avoidance of unpredictable, dynamically constrained obstacles". In: *Autonomous Robots* 32.3 (2012), pp. 227–242.

[Yan18] I. Yang. "A dynamic game approach to distributionally robust safety specifications for stochastic systems". en. In: *Automatica* 94 (Aug. 2018), pp. 94–101.