Stochastic Predictive Control for Legged Robots

Dissertation

der Mathematisch-Naturwissenschaftlichen Fakultät

der Eberhard Karls Universität Tübingen

zur Erlangung des Grades eines

Doktors der Naturwissenschaften

(Dr. rer. nat.)

vorgelegt von

M.Sc. Ahmad Gazar

aus Geburtsort

Monufia, Ägypten

Tübingen

2023

Gedruckt mit Genehmigung der Mathematisch-Naturwissenschaftlichen Fakultät der Eberhard Karls Universität Tübingen.

Tag der mündlichen Qualifikation:	05.12.2023
Dekan:	Prof. Dr. Thilo Stehle
1. Berichterstatter:	Prof. Dr. Ludovic Righetti
2. Berichterstatter:	Prof. Dr. Georg Martius

To Franzi,
this wouldn't be out without you.

Preface

Humans show distinguished capabilities in decision-making. Whether planning for daily commuting routes, deciding on the next move in a chess game, or shooting a rocket in space, we face the same conundrum of having to make hundreds of present decisions in the face of uncertain events in the future. By incorporating such uncertainties in the decision-making process, one can act with a sense of rationality given the information available about the world. Despite the long history of academic effort formulating this problem in the fields of operational research, numerical optimization, and control theory, planning under uncertainty persists to be a challenging problem in robotics and remains an active area of research.

One of the main challenges of planning under uncertainty stems from the computational intractability of the underlying mathematical programs given the limited computational resources against the realm of infinite uncertain future outcomes. Moreover, most of the interesting planning problems need to be solved in real-time, which adds to the complexity of the problem. For instance, If I am (an amateur chess player) to play a chess game against Garry Kasparov [1] where he has to wait for me 30 mins at a time during my turns to simulate as many scenarios as possible in the future to maximize my winning chances, not only it would be the most boring chess match ever played, but my limited computed scenarios might as well not be sufficient enough to win Kasparov. Imagine if we could optimize New York City's subway network for hours at a time to compute the optimal cycle of train routes and their frequencies, given the uncertainty of the number of daily commuters over all the network stations. This would be great except for the fact that New Yorkers will stop using the subway as they won't accept such optimization delay time before announcing the next cycle of routes for their commutes. Even if they would, it would be an outdated forecast already.

Another important aspect in planning under uncertainty constitutes satisfying a set of constraints along with a desired objective. For example, a robot navigating around a museum needs to approach randomly walking visitors to assist them, while ensuring a collision-free trajectory based on a robustness measure in the presence of randomly walking visitors around the museum. The choice of a robustness measure

[1] Garry Kasparov

for satisfying constraints is directly intertwined with the optimality of achieving a task. This is not an easy task, since an overestimation of future uncertainties can lead to a conservative behavior that deteriorates the task objective of a robot, while a very relaxed one can hinder safety by violating the planning constraints. Consider the case of a self-driving car on a highway that keeps skipping all exits on its way to the airport to guarantee a worst-case separation distance measure to avoid colliding with other cars. Although this measure achieves perfect robustness, it fails to achieve main the objective of getting to the airport in time by deteriorating the objective performance. On the other hand imagine the complement of this scenario, where a self-driving car arrives at the airport on time after making a couple of collisions with other cars to achieve its objective.

Designing robust trajectory optimizers for legged robots is a very challenging task. This is because of the task complexity of locomotion, and the unstable nature of the non-smooth under-actuated dynamics, which needs to be stabilized by making contact with the environment that respects the physics constraints. Due to this complexity, it became a common practice in the legged locomotion community to design sub-optimal robustness measures for constraint satisfaction either heuristically or using worst-case estimation of uncertainties to bypass the curse of computation using fast re-planning. Not only does this deteriorate the performance of the control policy as mentioned above, but we also argue that high-frequency re-planning is not helpful in the case of legged robots. This is due to the hybrid nature of the fast-switching dynamics of walking robots. Consider a legged robot climbing stairs while losing a step due to an external push or wrong estimates from its sensors. In this scenario, initiating a contact force in the air when it's supposed to be swinging its feet forward, can only make things worse due to the non-causality of the control policy. This is similar to a scenario of a dog wagging its tail, which does not get better even when the dog does it faster. We claim that anticipating such uncertainties during real-time planning can help achieve more robust locomotion behaviors for legged robots.

This thesis attempts to overcome the aforementioned challenges by designing a tractable real-time robust feedback control policy for legged robots subject to additive and parametric stochastic uncertainties on the model dynamics. We use tools inspired by stochastic model predictive control and chance-constraint programming to design uncertainty-aware constraints in a systematic non-conservative fashion for trajectory optimization of legged robots as a robustness measure. The final policy is designed as a trade-off between robustness and performance.

Acknowledgments

My deepest gratitude goes to my PhD advisors Ludovic Righetti and Andrea Del Prete for their guidance and support to me during my PhD.

Ludovic does not only resemble his known image as a rigorous scientist with very inspiring ideas but also a truly ethical and compassionate person as well. He managed to create an unparalleled free working environment (in the absolute sense) for his students to be their best selves. This fostered my scientific curiosity to pursue ideas with full autonomy as I've always imagined academia ought to be. Despite being in two different countries for most of my PhD, he has been extremely generous with his time with me, and unconditionally supported my scientific and personal goals without compromises. Thank you, Ludo for giving me the opportunity to be part of Movement Generation and Control and Machines in Motion. Working in your proximity has been truly humbling, and I strive to imitate your scientific image and ethics in my future endeavors.

Despite leaving for Italy after his professorship, Andrea has always spared the time for me to offer his help and discuss science even if he is traveling, on vacation, or even cutting from his family time to give me fast feedback. Andrea taught me scientific rigor, critical thinking, and challenging my ideas to become a better scientist. His discussions and sharp suggestions during our regular meetings have been invaluable during my PhD. Thank you Andrea for hosting me in Italy for 6 months, and for your genuine care about my PhD trajectory from day one. It has been very pleasurable working with you. You taught me a lot.

Majid Khadiv taught me everything about legged robotics. Thank you for the long hours you spend with me on the board discussing new ideas, deriving math, reading infinite papers, and more importantly the long coffee breaks. Not only Majid is a true collaborator in a scientific sense, but I have also been quite lucky to have his friendship and support during hard times in my PhD. His humbleness and altruism to share his knowledge and time with everyone is truly contagious. I could not have wished for a better person to share an office with all over the years.

I would like to thank the members of Movement Generation and Control, and Machines in Motion for sharing scientific discussions, shareable codes, comments, and feedback on my scientific work throughout my PhD. I learned a lot from all

of you by working in your vicinity. Thanks to Bilal and Sarah for their valuable academic advice and fun times inside and outside the lab at Max-Planck. My visiting time in NYC has been very enjoyable scientifically, personally, and politically thanks to Armand, Sébastien, and Huaijang.

I've been lucky to form very special friendships along my journey at Max-Planck. Manuel, your friendship has been truly precious to me and survived the test of time. Your compassion, support during hardships, and our endless intellectual and fun discussions have been a great catalyst. Thanks for your feedback on my papers as well. Cristina, I couldn't imagine my time at Max-Planck without your friendship and support. Thank you for keeping me company over the years, the hilarious moments mixed with infinite political discussions, and your true uncompromising honesty. To Laura, Nico, and Francesco, I am very lucky to have met you in Tuebingen. Thank you for your honest friendship. Without you, the journey would have been very difficult and boring. To Amr, I can't express my gratitude for your keen and caring advice. It has been and will always be truly pleasurable being your friend. To Nourhan, thanks for keeping your kind heart and unconditional supportive company to me despite the space-time difference over the years. To Patrick, thanks for making the world a better place by having people around like you. Your optimism will always inspire me.

To Lotte and Frede, your love and support have been very precious to me. I can't express how much I value your company as time passes.

To my family; **أمي وأبي، أشكركم على حبكم ودعمكم غير المشروط لي على مر السنين خلال رحلتي الأكاديمية، على الرغم من تعقيدات شخصيتي. أتمنى أن أكون قد جعلتكما فخورين بي وبالشخص الذي تطورت لأكون عليه. إلى إخوتي، أشكركم على وقوفكم بجانبي في أصعب أوقاتي، وتقديركم لاختفائي، وعدم تخصيص الوقت الكافي لكما حتى وصلت إلى كتابة هذه الأطروحة.**

Ahmad Gazar,
New York City, October 17, 2023

Contents

List of Figures

List of Tables

List of Abbreviations

MPC	Model Predictive Control
LMPC	Linear Model Predictive Control
RMPC	Robust Model Predictive Control
SMPC	Stochastic Model Predictive Control
NMPC	Nonlinear Model Predictive Control
HNMPC	Heuristic Nonlinear Model Predictive Control
SNMPC	Stochastic Nonlinear Model Predictive Control
OCP	Optimal Control Problem
ROC	Robust Optimal Control
SOC	Stochastic Optimal Control
LQR	Linear Quadratic Regulator
iLQR	Iterative Linear Quadratic Regulator
LGQ	Linear Quadratic Gaussian
DDP	Differential Dynamic Programming
QP	Quadratic Program
QCQP	Quadratic Constrained Quadratic Program
CoM	Center of Mass
CoP	Center of Pressure
LIPM	Linear Inverted Pendulem Model
SRGD	Single Rigid Body
TSID	Task-Space Inverse Dynamics
SCP	Sequential Convex Programming
SQP	Sequential Quadratic Programming
RPI	Robust Positive Invariant
mRPI	Minimal Robust Positive Invariant
PDF	Probability Density Function
CDF	Cumulative Density Function
gPC	Generalized Polynomial Chaos
KKT	Karush-Kuhn-Tucker

List of Symbols

$\boldsymbol{A}^\top$	Transpose of matrix A
$\boldsymbol{A}^{-1}$	Inverse of matrix A
$\boldsymbol{x}_{t+i\|t}$	Value of $\boldsymbol{x}$ at the future time $t+i$ predicted at time t
$\boldsymbol{x} \sim \mathcal{Q}$	A random variable $\boldsymbol{x}$ following a distribution $\mathcal{Q}$
$\mathbb{E}[\boldsymbol{x}]$	Expected value of $\boldsymbol{x}$
$\boldsymbol{\Sigma}_{\boldsymbol{x}} \triangleq \mathbb{E}[(\boldsymbol{x}-\mathbb{E}[\boldsymbol{x}])(\boldsymbol{x}-\mathbb{E}[\boldsymbol{x}])^\top]$	Covariance of $\boldsymbol{x}$

Operators and Functions

$\mathcal{A} \ominus \mathcal{B} = \{a \in \mathcal{A} \,\vert\, a + b \in \mathcal{A},\, \forall b \in \mathcal{B}\}$	Pontryagin set difference
$\mathcal{A} \oplus \mathcal{B} = \{a + b \,\vert\, a \in \mathcal{A},\, b \in \mathcal{B}\}$	Minkowski set sum
$\mathbf{diag}(\boldsymbol{x})$	Diagonal matrix constructed from vector $\boldsymbol{x}$
$\otimes$	Quaternion cross product
$\vert . \vert$	Element-wise absolute norm
$\Vert . \Vert$	l_2 norm
$\Vert . \Vert_p$	Weighted l_2 norm

Groups and Sets

$\mathbb{R}^{n \times m}$	Set of real matrices with n rows and m columns
$\mathbb{SO}(3)$	Special Orthonormal group
$\mathbb{SE}(3)$	Special Euclidean group

Others

$\mathbb{I}_n$	Identity matrix with size $n \times n$
$\mathbf{0}$	Zero matrix with appropriate dimensions

Part I

Background

Chapter 1

Introduction

1.1 Challenges in Locomotion of Legged Robots

Trajectory optimization has become a dominant paradigm for planning and control of legged robots [75, 86, 117, 17, 112]. Classically, a walking robot needs to satisfy a set of competing control objectives like moving towards some goal state with a desired velocity by exploiting its contacts with the environment subject to a set of constraints that respect the physics of the environment and the robot's physical capabilities. This problem is challenging for robots in contact with the environment due to the hybrid nature of their under-actuated dynamics that need to be stabilized through constrained contact forces at desired contact locations with the environment [90]. Model Predictive Control (MPC) has been a favorable tool of choice for trajectory optimization as it exploits the causal structure of the rolled-out dynamics while guaranteeing constraint satisfaction [89, 88].

Despite the inherent robustness of MPC for disturbance rejection through high-frequency re-planning [36, 51, 25, 66, 72], dealing with persistent disturbances remains critical for successful execution of agile motions for legged robots [33, 32]. Such disturbances are inevitable due to estimation errors, model mismatches between models used in planning and real-robot models, or imperfect controls. The accumulation of such uncertainties along with the impact disturbances during contact initiation can cause the risk of saturating or violating some of the physical constraints of the robot, which in turn causes failed robot motions [26, 109].

Another challenge facing MPC for legged robots is the limit of computation that can be performed in real-time. Depending on the task and model complexities, the optimization problem can be hard to converge in time before the next planning cycle [111, 58, 18]. This induces sub-optimal control policies that can misplace end-effectors at unintended contact locations, leading to failed motions [32]. We also argue that high-frequency re-planning is sometimes not helpful in robot locomotion.

This is due to the hybrid nature of the fast-switching dynamics of walking robots. For example, a robot climbs stairs, while losing a step due to an external push or wrong estimates from its sensors. In this scenario, initiating a contact force in the air when it's supposed to be swinging its feet forward, can only make things worse due to the non-causality of the control policy. This is similar to a scenario of a dog wagging its tail, which does not get better even when the dog does it faster. Thus, anticipating such uncertainties is a necessity in real-time planning, and can help achieve more robust locomotion behaviors for safe trajectory optimization [39, 28].

Although planning under uncertainty has been studied in portfolio optimization [64], numerical optimization [78, 97], and control theory [96, 53, 12], much of this theory did not transfer fully to trajectory optimization of robotic systems. This is because the task objectives for robotic systems are rather complicated, and are not easily reducible to classic control objectives such as simple regulation or tracking tasks. Moreover, till the moment of writing this, there is no unifying theory for robots to assess risks [61]. For instance, it's not clear what uncertainty measure a robot should take when it opens a door. what about when it's cooking or climbing stairs? Moreover, a successful robot plan does not depend solely on a performance metric, but should also provide some closed-loop robustness guarantees for constraint satisfaction that consider such uncertainty measures [3]. This requires the robot to be able to predict its uncertainties in the future and design a feedback control policy that counteracts such uncertainties to avoid constraint violations. This problem is computationally intractable in general as the controller needs to react to infinite possibilities of uncertainties, which is computationally intractable. Striking the right balance between optimally achieving a task while robustly satisfying constraints in uncertain environments is a challenging task in robotics. This is because both tasks can be conflicting. An overestimation of future uncertainties can lead to a conservative behavior that deteriorates the task objective of a robot, while a very relaxed one can hinder safety by violating the planning constraints [34].

1.2 Robust and Stochastic Optimal Control

A common way to deal with uncertainties is through Robust Optimal Control (ROC)/$\mathcal{H}_\infty$ control that attempts to solve minimax type of problems [93]. Minimax controllers are pessimistic as they deal with worst-case disturbance realizations, which is quite conservative and can lead to infeasibilities of the OCP [96]. A tractable approach to minimax is *tube-based Robust Model Predictive Control* (RMPC) [71], where an online nominal auxiliary controller along with a Robust Invariant Set (RPI) designed offline to contain all disturbed trajectories that guarantee robust constraint satisfaction of all trajectories inside the tube subject to

a bounded disturbance set. Similar ideas were explored for nonlinear polynomial systems by designing offline pre-planned "funnels of trajectories" using Sum-of-Squares(SoS), and tracking controllers based on sensor feedback [62]. Similarly, Manchester and Kuindersma [63] developed a robust controller with ellipsoidal disturbance sets and a local LQR feedback control law around a reference trajectory.

ROC tends to sacrifice optimality for attaining robust behavior against worst-case scenarios at all times. Another alternative to circumvent this conservativeness is Stochastic Optimal Control (SOC), which falls under the category of $\mathcal{H}_2$ Controllers. Different from ROC, SOC deals with random uncertainties in the process model with known probability distributions. The goal is then to design an expected cost based on the statistical information of the probability distribution of such uncertainties [54]. The most well-studied case in this category of problems is the Linear Quadratic Gaussian (LQG) control problem, which assumes additive Gaussian noise on the dynamics along with a Kalman filter estimator that observes the evolution of the state [55]. In this special case, the control problem reduces to the classic Linear Quadratic Regulator (LQR) case based on the separation principle, where the estimation and control problems are decoupled [99, 118]. Due to the unboundedness of the uncertainty description based on the stochastic distribution assumption, constraint satisfaction can not be guaranteed. One way to deal with recursive constraint satisfaction (recursive feasibility) in Stochastic MPC (SMPC) is to design stochastic tubes known as Probabilistic Invariant Sets (PISs) with specific cross-section shapes (e.g. ellipsoidal or polytopic) subject to stochastic disturbances with bounded distribution support [13, 59]. Tube-based SMPC approaches can share a similar degree of conservatism as tube-based RMPC since they assume bounded support of the uncertainty distribution. It has been shown that there is a correspondence between probabilistic Invariant sets (PIS) with ellipsoidal and polytopic confidence bounds and RIS [43].

One way to reduce the conservatism of tube-based RMPC and SMPC approaches, is to consider explicit unbounded probability distributions in the design of constraints by relaxing the constraint satisfaction to not exceed a permitted probability level, which is known as *Chance-Constraints* [78]. Chance-constrained SMPC offers a trade-off between the control performance (described by the cost function) and the robustness of constraint violations. This means that the performance can be achieved by operating the system closer to its state constraints, which comes at the cost of increased risk of constraint violation in a stochastic setting. As the state variance increases, the state must be moved farther away from the constraints to avoid the risk of violation. Hence, a central component of chance-constrained SMPC is predicting the state variances, such that the state can accordingly *back-off* from its constraints by an appropriate safety margin [40]. Depending on how critical the task is, the user can tune the desired probability

level between the two extremes of *almost* hard constraint satisfaction (as in RMPC) and complete negligence of disturbances (as in nominal MPC). This flexibility becomes very practical since a legged robot needs to move in dynamic environments where some of the constraints can be more critical than others. For example, when moving through a narrow doorway or walking in a crowd [20], the robot needs to reduce the sway motion of its Center of Mass (CoM) to reduce the probability of collision. However, for walking on challenging terrains with partial footholds [115], the robot has to bring the foot Center of Pressure (CoP) as close as possible to the center of the contact area. Many other tasks can be considered somewhere between those situations [34]. To this end, SMPC can be a powerful and systematic tool for dealing with constraint satisfaction in different environments and tasks. Moreover, small errors are typically more likely to occur in practice. It might therefore be more appropriate to explicitly consider the distribution of disturbances instead of treating all of them equally as in RMPC, which leads to a conservative behavior.

The work presented in the thesis aligns with the same philosophy and proposes to use tools from SMPC and chance-constraint programming for safe trajectory optimization of legged robots subject to stochastic additive and parametric uncertainties, and state and control chance-constraints [34, 33, 32]. We rely on the common assumption of Gaussian uncertainties for the analytic propagation of uncertainties in a receding horizon fashion inside the optimization problem.

1.3 Planning Under Uncertainty in Legged Robotics

Linear MPC has been used extensively as a prime candidate for generating a wide range of feasible reference walking motions for walking robots [41, 98, 113, 49, 24]. However, the theoretical guarantees associated with linear MPC like constraint satisfaction can be easily lost, due to external disturbances or the discrepancy between the nonlinear dynamics of the robot and the linearized model used in control. Approaches in [10, 22] studied how to account for the bounded error in constraint satisfaction due to the approximation of the nonlinear center of mass (CoM) dynamics, and Bohórquez et al.[7] investigated the nonlinear constraints due to step-timing adaptation. However, they do not account for the closed-loop tracking errors due to disturbances, and they provide no robustness guarantees for constraint satisfaction in the presence of different disturbances, which is critical for generating safe walking motions.

Linear Robust MPC (RMPC) schemes have been extensively studied in the control literature [67, 19, 71] to deal with persistent deterministic disturbances on the model dynamics. Villa et al. [109] used the well-known tube-based RMPC approach originally developed in [71] for generating robust walking motions for humanoid robots, taking into account the effects of additive compact polytopic uncertainties

on the dynamics. Using a state feedback control policy and a pre-stabilizing choice of static dead-beat gains, they showed that constraints are guaranteed to be satisfied for all disturbance realizations inside the disturbance set. Pandala et al.[83] developed a linear RMPC framework along with a deep reinforcement learning module to learn the vertices of a polytopic disturbance to bridge the gap between the reduced-order Single Rigid Body (SRBD) model for quadrupedal robots, and the full-body model. Recently, Xu et al. [119] solved linear minimax robust MPC for the SRGD model subject to process model and friction constraints uncertainties. By exploiting the structure of the uncertainties, they were able to reformulate the minimax problem as a convex Quadratically Constrained Quadratic Program (QCQP) that can be solved in real-time on a quadruped robot. A drawback of RMPC is that the constraints are designed to accommodate for the worst-case disturbance, which is quite conservative and sacrifices the task performance (optimality) to guarantee hard constraint satisfaction.

In order to relax the conservativeness of RMPC, SMPC [12, 40, 29, 59] exploits the underlying probability distribution of the disturbance realizations. Furthermore, SMPC offers a flexible framework by accounting for chance constraints, where constraints are expected to be satisfied within a desired probability level [78, 80]. Although linear models like the Linear Inverted Pendulum Model (LIPM) allow the application of linear SMPC approaches [34], it limits the range of agile motions and relevant uncertainties to be considered for legged robots, and their effect on contact location constraint satisfaction.

For optimizing whole-body motions, indirect methods like iLQR/Differential Dynamic Programming (DDP) [68] have become a popular choice in the robotics community [104, 65]. To incorporate uncertainties in DDP formulations, Morimoto et al. considered a minimax DDP for simple bipedal walking dynamics subject to additive disturbances on the viscous friction of the robot joints [76]. Like RMPC, minimax approaches tend to be overly conservative as they deal with worst-case disturbances, which sacrifice the task performance and can lead to infeasibilities. Recently [39, 38, 46] used risk-sensitive DDP that accounts for process and measurement uncertainties. Del Prete et al. [26] considered joint-torque chance-constraints robustness inside Task-Space Inverse Dynamics (TSID) for whole-body control of humanoid robots subject to joint torque uncertainties. Different from receding horizon approaches, TSID is an instantaneous controller that is myopic to long-horizon tasks like locomotion. Other lines of work resorted to sampling-based methods to approximate the stochastic optimal control problem (OCP). For instance, Mordatch et al. used an ensemble of perturbed models that allowed them to transfer the control policy to a humanoid robot [74]. Yeganegi et al. [120] used Bayesian optimization to learn cost function weights to achieve robust walking motions under different uncertainties. Suh et al . [101] used randomized smoothing

for obtaining derivative-free gradients for solving the OCP as an alternative to domain randomization. However, sampling-based approaches are computationally expensive for real-time MPC applications of high-dimensional robotic systems. Despite the different risk measures adopted in the above trajectory optimization approaches, they do not include constraints in their formulations, which is essential for safe trajectory optimization.

Other approaches incorporated uncertainties through contact by solving a Stochastic Linear Complementarity Problem (SLCP). For example, Tassa et al. [103] solved an SLCP to avoid the discontinuities of the complementarity problem. This allowed them to optimize smoothly through contacts by offering a trade-off between contact complementarity accuracy, and the feasibility of the problem. Despite the success of DDP approaches, they do not consider the effect of uncertainties on constraint satisfaction, which is crucial for robotic systems. Finally, Drnach et al. used a direct contact-implicit approach to solve an SLCP with chance-constraints [28]. Due to the nature of the non-smooth mixed-integer problem of contact-implicit approaches, they are hard to solve and are best suited for offline trajectory optimization.

1.4 Contributions

Some of the limitations of the previous approaches are: 1) they do not consider explicitly the effect of uncertainty on constraint satisfaction, which is the case in most aforementioned DDP approaches. 2) Contact-implicit approaches are usually hard to tune and get easily stuck in local minima, which limits their applicability for MPC. 3) Unlike stochastic trajectory optimization, robust approaches are conservative as they sacrifice performance for safety. 4) Sampling-based methods are computationally expensive to be considered for real-time planning of highly dimensional robotic systems, and they require a big engineering effort during parallelization. This thesis attempts to overcome the above limitations by designing chance-constrained SMPCs for legged robots under additive and parametric stochastic uncertainties in chapter 3-chapter 5.

Our contributions in chapter 3 are [34]:

- We introduce linear SMPC using a linear inverted pendulum model (LIPM) to generate stable walking, taking into account stochastic model uncertainty subject to individual chance constraints.

- We analyze the robustness of SMPC to worst-case disturbances, drawing an interesting connection between robust and stochastic MPC, and highlight their fundamental difference.

- We compare SMPC, RMPC, and nominal MPC in terms of robustness (constraints satisfaction) and performance. Our tests focus on stochastic bounded disturbances (generated with a truncated Gaussian distribution), which are a good approximation of real disturbances as joint torque tracking errors [26]. We empirically show that SMPC can achieve hard constraint satisfaction while being significantly less conservative than RMPC [1].

Our contributions in chapter 4 are [33]:

- We propose to use stochastic trajectory optimization using Sequential Convex Programming (SCP) for generating robust centroidal momentum trajectories subject to additive uncertainties in the dynamics, as well as parametric contact position uncertainties.

- By considering chance constraints on the friction cones, this is the first work (to the best of our knowledge) that designs controllers for legged robots that generate robust force trajectories subject to contact-location uncertainty. In the same spirit as [42], we propose a whole-body trajectory planning framework that alternates between whole-body and centroidal momentum planning. Contrary to [42]—but similarly to [11]—we use whole-body DDP instead of a kinematics optimization, and use the solution to warm-start the stochastic SCP centroidal momentum solver. The resulting (robust) momentum trajectories are later used by the whole-body DDP again to track the resulting robust centroidal trajectories and contact force trajectories. This way, we make sure that the generated momentum trajectories are robust while being consistent with the robot whole-body dynamics.

- Finally, we run 400 Monte-Carlo simulations on the open-source quadruped robot, Solo, [37] in a Pybullet simulation environment [21] for dynamic trotting and bounding gaits while applying different disturbances. We show that Stochastic trajectory optimization can complete all the motions safely while reducing feet slippage, and achieving better centroidal tracking performance over the deterministic planning [2].

We highlight the main result in chapter 5 as the first work on receding horizon whole-body Stochastic Nonlinear MPC (SNMPC) that deals with contact-location uncertainty for agile motions of legged robots. Our contributions are [32]:

- We solve a stochastic kino-dynamic whole-body trajectory optimization subject to additive uncertainties in the dynamics. Contrary to our previous work

[1] submission video.
[2] submission video.

on stochastic centroidal momentum trajectory optimization [33], we optimize both the centroidal dynamics and the full robot kinematics, which allows us to model uncertainties on the optimized contact locations in a receding horizon fashion rather than on fixed contact locations with fixed parametric contact location uncertainties.

- We design contact location chance constraints inside an approximate real-time SQP-type iteration. This is less conservative than considering worst-case disturbance in robust optimization, where constraints are to be satisfied for all possible realizations. Instead, we satisfy constraints in a probabilistic sense, while maintaining the same computational complexity as NMPC without degrading the performance.

- We compared SNMPC against NMPC by running extensive Monte-Carlo simulations of the quadruped robot Solo for dynamic trotting and bounding gaits on a challenging non-coplanar terrain. Furthermore, We compared the robustness induced by SNMPC against a heuristic-based NMPC (HNMPC), where the contact location constraints were shrunk by hand using a heuristic safety margin. Our results show that SNMPC was able to perform all motions safely with 100% success rate, while NMPC and HNMPC failed 48.3% and 47.6% of the time respectively [3].

1.5 Publications

The following papers were published/pending review during the course of my PhD, which chapter 3, chapter 4, and chapter 5 of the thesis builds upon:

published Ahmad Gazar, Majid Khadiv, Andrea Del Prete, and Ludovic Righetti. Stochastic and robust mpc for bipedal locomotion: A comparative study on robustness and performance. In *2020 IEEE-RAS 20th International Conference on Humanoid Robots (Humanoids)*, pages 61–68, 2021

published Ahmad Gazar, Majid Khadiv, Sébastien Kleff, Andrea Del Prete, and Ludovic Righetti. Nonlinear stochastic trajectory optimization for centroidal momentum motion generation of legged robots. In *Robotics Research*, pages 420–435, 2023

under review Ahmad Gazar, Majid Khadiv, Andrea Del Prete, and Ludovic Righetti. Multi-contact stochastic predictive control for legged robots with contact locations uncertainty. *arXiv preprint arXiv:2309.04469*, 2023

[3] submission video.

A related publication is

published Shahram Khorshidi, Ahmad Gazar, Nicholas Rotella, Maximilien Naveau, Ludovic Righetti, Maren Bennewitz, and Majid Khadiv. On the use of torque measurement in centroidal state estimation. In *2023 IEEE International Conference on Robotics and Automation (ICRA)*, pages 9931–9937, 2023

1.6 Outline

This thesis is organized as follows: chapter 1 starts by introducing some of the challenges in legged robotics, classic control-theoretic methods for planning under uncertainty, and finally the state-of-the-art methods in deterministic and robust trajectory optimization for legged robots under different uncertainties descriptions and their limitations. We summarize the differences between those methods in terms of the underlying model complexity used in planning—linear vs nonlinear. Furthermore, we differentiate between offline vs online (receding horizon MPC) methods, the underlying robustness measures adopted in planning (robust vs stochastic approaches), as well as constrained vs unconstrained approaches. Then, we present the contributions of the thesis as a step towards overcoming some of the shortcomings of the presented state-of-the-art methods, along with the list of publications published/submitted during this PhD.

In chapter 2, we present the robot models used in this thesis along with their deterministic OCP formulations for trajectory optimization of legged robots.

The first contribution is presented in chapter 3. First, we introduce a classic tube-based RMPC approach applied to bipedal locomotion under additive disturbances on the LIPM dynamics. Then, we introduce SMPC under stochastic uncertainty with individual state and control chance constraints and compare its objectives against RMPC and deterministic linear MPC for CoM trajectory generation of a humanoid robot passing throw a narrow hall-way in a safety-critical scenario. We compare SMPC, RMPC, and nominal MPC in terms of robustness (constraints satisfaction) and performance, and highlight the fundamental difference of between SMPC and RMPC. We make the argument that SMPC provides a more optimal and flexible robustness measure for the trajectory optimization of legged robots.

In chapter 4, we extend the approach to stochastic trajectory optimization for centroidal momentum motion generation of legged robots subject to additive and parametric contact-location uncertainties on the centroidal momentum dynamics, and friction pyramid joint chance constraints. Moreover, we introduce a trajectory optimization framework that alternates between stochastic centroidal trajectory optimization, and whole-body DDP for generating robust whole-body motions under different uncertainties. We also highlight the benefits of this approach over

deterministic planning while maintaining the same computational complexity on a quadrupedal robot performing agile motions on challenging uncertain terrains.

chapter 5 presents our latest submission, and the main result of the thesis. We present a nonlinear SMPC algorithm applied in real-time for whole-body trajectory optimization of legged robots subject to additive stochastic uncertainties and contact-location joint chance constraints. We showed that whole-body chance-constrained SNMPC with a real-time iteration scheme generates robust motions for agile motions over the classic NMPC, with almost the same performance. We also compared empirically chance-constraints tightening against the commonly adopted heuristic-based NMPC in legged robotics, where constraints are tightened with a fixed safety margin using trial and error. We showed that SNMPC provides a formal, generic, and flexible way of achieving robustness without sacrificing the computational tractability of the OCP.

Finally, we summarize the limitations of the presented work in the thesis and propose some future directions to overcome such limitations in chapter 6.

Chapter 2

Deterministic Optimal Control for Legged Robots

The full-body dynamics of a floating-base robot in contact with the environment can be derived using Euler-Lagrange equations of motion as follows [113]:

$$\boldsymbol{M}(\boldsymbol{q})\ddot{\boldsymbol{q}} + \boldsymbol{h}(\boldsymbol{q}, \dot{\boldsymbol{q}}) = \sum_{i=1}^{n_c} \boldsymbol{J}_{e,i}^{\top}(\boldsymbol{q})\boldsymbol{\lambda}_{e,i} + \boldsymbol{S}^{\top}\boldsymbol{\tau}_q. \tag{2.1}$$

The generalized robot position $\boldsymbol{q} = \left[\boldsymbol{x}_b^{\top}, \boldsymbol{\theta}_j^{\top}\right]^{\top} \in \mathbb{SE}(3) \times \mathbb{R}^{n_j}$ represents the robot's floating base pose, and joint positions respectively. represents the generalized robot position characterizing the robot's floating base pose (position and orientation) w.r.t. an inertial frame $\mathcal{I}$, and the joint positions respectively. $\boldsymbol{M}(\boldsymbol{q}) \in \mathbb{R}^{(6+n_j)\times(6+n_j)}$ denotes the inertia matrix, and $\boldsymbol{h}(\boldsymbol{q}, \dot{\boldsymbol{q}}) \in \mathbb{R}^{6+n_j}$ is the vector capturing the Coriolis, centrifugal, gravity and joint friction forces. $\boldsymbol{J}_{e,i}$ is the associated jacobian of the i-th end-effector wrench $\boldsymbol{\lambda}_{e,i}$ acting on the environment. Finally, $\boldsymbol{S} = \left[\boldsymbol{0}_{(n_j\times 6)}, \mathbb{I}_{n_j}\right]$ is the selector matrix of the actuated joint torques $\boldsymbol{\tau}_q$.

2.1 Nominal Linear MPC for LIPM

The dynamics of the CoM of a walking robot, under the assumption of rigid contacts with a flat ground, can be modeled as follows [114]:

$$p^{x,y} = c^{x,y} - \frac{m_{tot}\, c^z\, \ddot{c}^{x,y} - S\dot{L}^{x,y}}{m_{tot}(\ddot{c}^z + g^z)}, \tag{2.2}$$

where $c \in \mathbb{R}$ denotes the CoM position in the lateral directions of motion ^{x,y}. The total mass of the robot is denoted by m_{tot}, the matrix $S = \left[\begin{smallmatrix} 0 & -1 \\ 1 & 0 \end{smallmatrix}\right]$ is a rotation

matrix, with CoP $p \in \mathbb{R}$ being constrained inside the convex hull of the contact points $\mathcal{U}$

$$p^{x,y} \in \mathcal{U}. \tag{2.3}$$

Under the assumption of constant CoM height c^z and constant angular momentum L, the dynamics (2.2) can be simplified to the LIPM, resulting in the following linear relationship between the CoM and the CoP

$$\ddot{c}^{x,y} = \omega_n^2(c^{x,y} - p^{x,y}), \tag{2.4}$$

where $\omega_n = \sqrt{\frac{g^z}{c^z}}$ represents the system's natural frequency, and g^z being the norm of the gravity vector along z. From now on, we will drop ^{x,y} superscripts for convenience. Consider the discrete-LTI dynamics (2.4) subject to state and control constraints:

$$x_{t+i+1} = Ax_{t+i} + Bu_{t+i}, \tag{2.5a}$$
$$x_{t+i+1} \in \mathcal{X}, \tag{2.5b}$$
$$u_{t+i} \in \mathcal{U}, \tag{2.5c}$$

where the state $x = \begin{bmatrix} c & \dot{c} \end{bmatrix}^\top \in \mathbb{R}^n$, with $n = 2$, and the control input $u = p \in \mathbb{R}^m$, with $m = 1$. $\mathcal{X}$ represents the set of linear kinematic constraints of the robot, like self-collision, maximum stride length, etc. MPC deals with solving the following optimal control problem (OCP) at every sampling time t:

$$\underset{\mathbf{u}}{\text{minimize}} \quad J_N(x_t, \mathbf{u}) \tag{2.6a}$$
$$\text{subject to}$$
$$x_{t+i+1|t} = Ax_{t+i|t} + Bu_{t+i|t}, \tag{2.6b}$$
$$x_{t+i+1|t} \in \mathcal{X}, \tag{2.6c}$$
$$u_{t+i|t} \in \mathcal{U}, \tag{2.6d}$$
$$x_{t|t} = x_t, \tag{2.6e}$$
$$i = 0, ..., N-1. \tag{2.6f}$$

$\mathbf{u} = \{u_{t|t}, u_{t+1|t}, ..., u_{t+N-1|t}\}$ denotes the control sequence along the prediction horizon N and $\mathbf{u}^*(x_t)$ is the minimizer of (2.6) given the current initial condition x_t. The above MPC scheme applies only the first control action $u^*_{t|t}(x_t)$ of the optimal open-loop control sequence. We avoided using terminal constraints (e.g capturability [52]) in our comparison, since to the best of our knowledge there is no systematic way for handling terminal constraints in SMPC as in nominal MPC

and RMPC. One of the options for generating viable reference walking trajectories using the above MPC scheme without terminal constraints is to minimize one of the CoM derivatives, adding it to the cost function J_N [113, 41, 114]. With a sufficiently long N a valid choice of the cost function in (2.6a) can be

$$J_N(x_t, \mathbf{u}) = \sum_{i=0}^{N-1} \alpha(\dot{c}_t^d - \dot{c}_{t+i|t})^2 + \beta(c_t^d - c_{t+i|t})^2 + \gamma(p_t^d - p_{t+i|t})^2. \tag{2.7}$$

c_t^d, and $\dot{c}_t^d$ represent the desired walking direction and velocity of the robot respectively. p_t^d denotes the desired CoP tracking position, which is usually chosen to be at the center of $\mathcal{U}$ for robustness. α, β and γ are user-defined weights.

2.2 Centroidal Nonlinear Trajectory Optimization

The full robot dynamics can be split (2.1) into its under-actuated and actuated parts;

$$\boldsymbol{M}^u(\boldsymbol{q})\ddot{\boldsymbol{q}} + \boldsymbol{h}^u(\boldsymbol{q}, \dot{\boldsymbol{q}}) = \sum_{i=1}^{n_c} \boldsymbol{J}_{e,i}^{u\top}(\boldsymbol{q})\boldsymbol{\lambda}_{e,i}, \tag{2.8a}$$

$$\boldsymbol{M}^a(\boldsymbol{q})\ddot{\boldsymbol{q}} + \boldsymbol{h}^a(\boldsymbol{q}, \dot{\boldsymbol{q}}) = \sum_{i=1}^{n_c} \boldsymbol{J}_{e,i}^{a\top}(\boldsymbol{q})\boldsymbol{\lambda}_{e,i} + \boldsymbol{\tau}_q. \tag{2.8b}$$

By writing down the floating-base dynamics for the CoM instead of the floating-base position, we obtain the following relationship between the centroidal momentum dynamics $\dot{\boldsymbol{h}}_{\mathcal{G}}$ and the generalized velocities $\dot{\boldsymbol{q}}$

$$\dot{\boldsymbol{h}}_{\mathcal{G}} = \begin{bmatrix} \dot{\boldsymbol{\kappa}} \\ \dot{\boldsymbol{l}} \end{bmatrix} = \dot{\boldsymbol{A}}_{\mathcal{G}}(\boldsymbol{q})\ddot{\boldsymbol{q}} + \dot{\boldsymbol{A}}_{\mathcal{G}}(\boldsymbol{q})\dot{\boldsymbol{q}}, \tag{2.9}$$

via the *Centroidal Momentum Matrix* (CMM) $\boldsymbol{A}_{\mathcal{G}} \in \mathbb{R}^{6\times(n+6)}$ [82]. The angular and linear momenta are denoted as $\boldsymbol{\kappa}$ and $\boldsymbol{l} \in \mathbb{R}^3$ respectively. Given (2.9), we are interested in planning desired centroidal momentum trajectories that satisfy the following Newton-Euler dynamics:

$$\dot{\boldsymbol{h}} = \begin{bmatrix} \sum_{i=1}^{n_c} (\boldsymbol{p}_{e,i} + \boldsymbol{R}_{e,i}^{x,y}\boldsymbol{\zeta}_{e,i} - \boldsymbol{c}) \times \boldsymbol{\lambda}_{e,i} + \boldsymbol{R}_{e,i}^{z}\tau_{e,i} \\ m\boldsymbol{g} + \sum_{i=1}^{n_c} \boldsymbol{\lambda}_{e,i} \end{bmatrix} \tag{2.10}$$

where $\boldsymbol{c} \in \mathbb{R}^3$ represents the robot's CoM, at which the total mass m of the robot is concentrated. $\boldsymbol{p}_{e,i} \in \mathbb{R}^3$ is now the i-th end-effector's contact position, with

$\boldsymbol{\zeta}_{e,i} \in \mathbb{R}^2$ being the local *Center of Pressure* (CoP). $\boldsymbol{\lambda}_{e,i} \in \mathbb{R}^3$ and $\tau_{e,i} \in \mathbb{R}$ represent end-effector's contact forces and torque for flat-footed robots, respectively. The rotation matrix $\boldsymbol{R}_{e,i} \in \mathbb{SO}(3)$ captures the contact normals mapping quantities from the i-th end-effector's frame to the inertial frame. Note that for point-footed robots, which we consider from now on, $\boldsymbol{\zeta}_{e,i}$ and $\tau_{e,i}$ are always null, but the same analysis still holds for flat-footed robots. First, we present the deterministic nonlinear discrete-time optimal control problem (OCP) for centroidal momentum trajectory optimization with fixed contact position and timing.

Problem 1. *Nominal Optimal Control Problem (NOCP)*

$$\underset{\boldsymbol{x},\boldsymbol{u}}{\text{minimize}} \quad l_f(\boldsymbol{x}_N) + \sum_{i=0}^{N-1} l(\boldsymbol{x}_k, \boldsymbol{u}_k) \tag{2.11a}$$

subject to

$$\begin{bmatrix} \boldsymbol{c}_{k+1} \\ \boldsymbol{l}_{k+1} \\ \boldsymbol{\kappa}_{k+1} \end{bmatrix} = \begin{bmatrix} \boldsymbol{c}_k + \frac{1}{m}\boldsymbol{l}_k \Delta_k \\ \boldsymbol{l}_k + m\boldsymbol{g}\Delta_k + \sum_{i=1}^{n_c} \boldsymbol{f}_{e,i_k} \Delta_k \\ \boldsymbol{\kappa}_k + \sum_{i=1}^{n_c} (\boldsymbol{p}_{e,i_k} - \boldsymbol{c}_k) \times \boldsymbol{f}_{e,i_k} \Delta_k \end{bmatrix}, \tag{2.11b}$$

$$-\mu \mathfrak{f}^z_{e,i_k} \leq \mathfrak{f}^x_{e,i_k} \leq \mu \mathfrak{f}^z_{e,i_k}, \quad \mathfrak{f}^z_{e,i_k} \geq 0, \tag{2.11c}$$

$$-\mu \mathfrak{f}^z_{e,i_k} \leq \mathfrak{f}^y_{e,i_k} \leq \mu \mathfrak{f}^z_{e,i_k}, \quad \mathfrak{f}^z_{e,i_k} \geq 0, \tag{2.11d}$$

$$|\boldsymbol{p}_{e,i_k} - \boldsymbol{c}_k| \leq \mathcal{L}^{\max}_{e,i}, \tag{2.11e}$$

$$\boldsymbol{x}_0 = \boldsymbol{x}(0), \tag{2.11f}$$

$$\boldsymbol{x}_f = \boldsymbol{x}(N), \tag{2.11g}$$

$$\forall k \in \{0, 1, .., N-1\}, \tag{2.11h}$$

where $\boldsymbol{x} = \{\boldsymbol{x}_0, \ldots, \boldsymbol{x}_N\}$ with $\boldsymbol{x}_k \in \mathbb{R}^9 = [\boldsymbol{c}_k, \boldsymbol{l}_k, \boldsymbol{\kappa}_k]^\top$, and $\boldsymbol{u} = \{\boldsymbol{u}_0, \ldots, \boldsymbol{u}_{N-1}\}$ with $\boldsymbol{u}_k \in \mathbb{R}^{3n_c} = [\boldsymbol{\lambda}_{e,1}, \ldots, \boldsymbol{\lambda}_{e,n_c}]^\top$ are the states and control optimizers along the control horizon N. The centroidal momentum dynamics are discretized with a time-step Δ_k using an explicit Euler integration scheme (2.11b), where (2.11f)-(2.11g) represents the initial and final conditions respectively. To avoid contact slippage, the local contact forces in the end-effector frame ($\mathfrak{f} = \boldsymbol{R}^T_{e,i_k} \boldsymbol{\lambda}_{e,i_k}$) are constrained inside the linearized friction cone constraints (2.11c)-(2.11d), where the static coefficient of friction is denoted as μ with the vertical component of the force being positive. Finally, the CoM is constrained to be within the leg length reachability limits (2.11e).

2.3 Kino-dynamic Nonlinear MPC (NMPC)

With the same spirit as [23], we are interested in planning kino-dynamic whole-body motions using centroidal momentum dynamics and full robot kinematics (2nd

order kinematics) as follows:

$$\underbrace{\frac{d}{dt}\begin{bmatrix}\boldsymbol{c}\\ \boldsymbol{l}\\ \boldsymbol{\kappa}\end{bmatrix} = \begin{bmatrix}\frac{1}{m}\boldsymbol{l}\\ m\boldsymbol{g} + \sum_{i=1}^{n_c}\boldsymbol{\lambda}_i\\ \sum_{i=1}^{n_c}\left(\mathbf{FK}_i(\tilde{\boldsymbol{q}}) - \boldsymbol{c}\right)\times\boldsymbol{\lambda}_i\end{bmatrix}}_{\text{Centroidal momentum dynamics}}, \tag{2.12a}$$

$$\underbrace{\frac{d}{dt}\begin{bmatrix}\boldsymbol{p}_b\\ \Delta\boldsymbol{q}_b\\ \boldsymbol{\theta}_j\\ \boldsymbol{v}_b\\ \boldsymbol{\omega}_b\\ \boldsymbol{v}_j\end{bmatrix} = \begin{bmatrix}\boldsymbol{v}_b\\ \boldsymbol{\omega}_b\\ \boldsymbol{v}_j\\ \boldsymbol{a}_b\\ \boldsymbol{\psi}_b\\ \boldsymbol{a}_j\end{bmatrix}, \quad \tilde{\boldsymbol{q}} \triangleq \begin{bmatrix}\boldsymbol{p}_b\\ \frac{1}{2}\boldsymbol{q}_{\text{ref}_b}\otimes\Delta\boldsymbol{q}_b\\ \boldsymbol{\theta}_j\end{bmatrix}}_{\text{Full robot kinematics}}. \tag{2.12b}$$

$\boldsymbol{c}\in\mathbb{R}^3$ represents the CoM of the robot, with m being the total mass of the robot subject to the gravity vector $\boldsymbol{g}$. The forward kinematics function $\mathbf{FK}_i(.):\mathbb{Q}\mapsto\mathbb{R}^3$ computes the i-th end-effector's contact position for a given robot configuration. For the simplicity of dynamics integration and constraints linearization later, we choose to optimize for the relative base orientation $\Delta\boldsymbol{q}_b$ w.r.t. an absolute base reference $\boldsymbol{q}_{\text{ref}_b}$ instead of $\boldsymbol{q}_b$ directly. $\boldsymbol{p}_b$ and $\boldsymbol{\omega}_b\in\mathbb{R}^3$ are the base linear position and velocity, while $\boldsymbol{\omega}_b$ and $\boldsymbol{\psi}_b\in\mathbb{R}^3$ are the base angular velocity and acceleration respectively. Finally, we *transcribe* the above continuous dynamics using direct collocation into the following MPC problem with pre-specified contact mode and timing Δ_k. The state and control optimizers at the k-th discretization step are $\boldsymbol{x}_k = \left[\boldsymbol{c}_k^\top, \boldsymbol{l}_k^\top, \boldsymbol{\kappa}_k^\top, \boldsymbol{p}_{b_k}^\top, \Delta\boldsymbol{q}_{b_k}^\top, \boldsymbol{\theta}_{j_k}^\top, \boldsymbol{v}_{b_k}^\top, \boldsymbol{\omega}_{b_k}^\top, \boldsymbol{v}_{j_k}^\top\right]^\top\in\mathbb{R}^n$, and $\boldsymbol{u}_k = \left[\boldsymbol{\lambda}_{i,k}^\top, \ldots, \boldsymbol{\lambda}_{n_c,k}^\top, \boldsymbol{a}_{b_k}^\top, \boldsymbol{\psi}_{b_k}^\top, \boldsymbol{a}_{j_k}^\top\right]^\top\in\mathbb{R}^m$ with $n = 21+6n_c$, and $m = 6+6n_c$ for point feet robots.

Problem 2. *Kino-Dynamic NMPC Problem*

$$\underset{\boldsymbol{X},\boldsymbol{U},\boldsymbol{S}}{\text{minimize}}\ \mathcal{L}_{\text{total}}(\boldsymbol{X},\boldsymbol{U},\boldsymbol{S}) \tag{2.13a}$$

$$\text{subject to} \tag{2.13b}$$

$$\boldsymbol{f}_{\text{impl}}(\boldsymbol{x}_k,\boldsymbol{x}_{k+1},\boldsymbol{u}_k) = \boldsymbol{0}, \tag{2.13c}$$

$$\boldsymbol{h}(\boldsymbol{x}_k,\boldsymbol{u}_k) + \boldsymbol{J}_{\text{sh}}\boldsymbol{s}_k \leq \boldsymbol{0}, \tag{2.13d}$$

$$-\boldsymbol{s}_k \leq \boldsymbol{0}, \tag{2.13e}$$

$$\boldsymbol{x}_0 - \boldsymbol{x}(t) = \boldsymbol{0}, \quad \forall k\in\{0,1,\ldots,N-1\}. \tag{2.13f}$$

where $\boldsymbol{X} \triangleq \{\boldsymbol{x}_0, \ldots, \boldsymbol{x}_N\}$ and $\boldsymbol{U} \triangleq \{\boldsymbol{u}_0, \ldots, \boldsymbol{u}_{N-1}\}$ are the states and control variables along the control horizon N. The implicit discrete dynamics $\boldsymbol{f}_{\text{impl}}(.) : \mathbb{R}^n \times \mathbb{R}^n \times \mathbb{R}^m \mapsto \mathbb{R}^n$ (2.13c) captures the kino-dynamic equality path constraints in (2.12) discretized using first-order implicit-Euler integration scheme, and transcribed using Gauss-Legendre collocation method. The remaining nonlinear path constraints $\boldsymbol{h}(.)$ are implemented softly to avoid infeasibilities of the OCP by introducing extra slack variables $\boldsymbol{S} \triangleq \{\boldsymbol{s}_0, \ldots, \boldsymbol{s}_N\}$, where $\boldsymbol{J}_{\text{sh}}$ selects the slack variable attached to the respective constraint. The constraints $\boldsymbol{h}(.)$ are described in detail in (2.14)-(2.17). At every receding horizon, the initial condition of the OCP is reset with the current measured state $\boldsymbol{x}(t)$ using constraint (2.13f). We enforce Kino-dynamic consistency between the (2.12a), and (2.12b) with the following constraints:

$$\boldsymbol{h}_{\text{kindyn}}(\boldsymbol{c}_k, \boldsymbol{h}_{\mathcal{G}_k}, \tilde{\boldsymbol{q}}_k) + \boldsymbol{s}_{\text{kindyn}_k} = \boldsymbol{0}, \tag{2.14a}$$

$$\boldsymbol{h}_{\text{kindyn}} \triangleq \begin{bmatrix} \boldsymbol{c}_k - \mathbf{COM}(\tilde{\boldsymbol{q}}_k), \\ \begin{bmatrix} \boldsymbol{\kappa}_k^\top, \boldsymbol{l}_k^\top \end{bmatrix}^\top - \boldsymbol{A}_{\mathcal{G}}(\tilde{\boldsymbol{q}}_k)\dot{\boldsymbol{q}} \end{bmatrix}, \tag{2.14b}$$

where $\mathbf{COM}(.) : \mathbb{Q} \mapsto \mathbb{R}^3$ function computes the center of mass of the robot for a given configuration, and $\boldsymbol{s}_{\text{kindyn}_k} \in \mathbb{R}^9$ are the slack variables associated with those kino-dynamic constraints. To avoid contact slippage, the tangential contact forces in the end-effector frame ($\mathfrak{f}_{i,k} = \boldsymbol{R}_{i,k}^\top \boldsymbol{\lambda}_{i,k}$) are constrained inside the friction cone

$$\gamma_{i,k} \,.\, \left[h_{\text{cone}_{i,k}}(\boldsymbol{\lambda}_{i,k}) + s_{\text{cone}_{i,k}} \leq 0 \right] \quad \gamma_{i,k} \in \mathcal{C}, \tag{2.15a}$$

$$h_{\text{cone}_{i,k}} \triangleq \sqrt{\mathfrak{f}_{x,i_k}^2 + \mathfrak{f}_{y,i_k}^2} - \mu \mathfrak{f}_{z,i_k}, \tag{2.15b}$$

where $\mathcal{C} = \{0, 1\}$. The contact mode (fixed apriori) $\gamma_{i,k} = 1$ when the i-th foot is in contact with the ground, and $\gamma_{i,k} = 0$ otherwise. The coefficient of friction is denoted by μ, and $s_{\text{cone}_{i,k}} \in \mathbb{R}$ is the slack variable associated with the friction cone constraint. During contact, the i-th end-effector position in the z-direction must be at the height of the contact surface $\mathcal{S}_{i,k}^z$, and be within the contact surface boundaries $\mathcal{S}_i^{x,y}$.

$$\gamma_{i,k} \,.\, \left[h_{\text{pos}_{i,k}}^z(\tilde{\boldsymbol{q}}_k) + s_{\text{pos}_{i,k}}^z = \mathcal{S}_{i,k}^z \right] \quad \gamma_{i,k} \in \mathcal{C}, \tag{2.16a}$$

$$\gamma_{i,k} \,.\, \left[\boldsymbol{h}_{\text{pos}_{i,k}}^{x,y}(\tilde{\boldsymbol{q}}_k) + \boldsymbol{s}_{\text{pos}_{i,k}}^{x,y} \in \mathcal{S}_{i,k}^{x,y} \right] \quad \gamma_{i,k} \in \mathcal{C}, \tag{2.16b}$$

where $\boldsymbol{h}_{\text{pos}_{i,k}} \triangleq \text{FK}_{i,k}(\tilde{\boldsymbol{q}}_k)$, and $\boldsymbol{s}_{\text{pos}_{i,k}} \in \mathbb{R}^3$ are the slack variables associated with the contact position constraints. For simplicity, we assume that $\mathcal{S}_{i,k} \in \mathbb{R}^3$ is a rectangular polytope. Finally, the end-effector velocities during contact are constrained to be zero by enforcing the holonomic constraint:

$$\gamma_{i,k} \,.\, \left[\boldsymbol{h}_{\text{vel}_{i,k}}(\tilde{\boldsymbol{q}}_k, \dot{\boldsymbol{q}}_k) + \boldsymbol{s}_{\text{vel}_{i,k}} = \boldsymbol{0} \right] \quad \gamma_{i,k} \in \mathcal{C}, \tag{2.17}$$

where $\boldsymbol{h}_{\mathrm{vel}_{i,k}} \triangleq \boldsymbol{J}_{i,k}(\tilde{\boldsymbol{q}}_k)\dot{\boldsymbol{q}}_k$, and $\boldsymbol{s}_{\mathrm{vel}_{i,k}} \in \mathbb{R}^3$ are the associated slack variables. In the above optimal control problem, we track a whole-body reference trajectory $\boldsymbol{x}_{\mathrm{r}}$ optimized apriori offline. The total cost in (2.13a) is split between the least-squares tracking cost $\mathcal{L}_{\mathrm{LS}}$, and the penalty cost $\mathcal{L}_{\mathrm{penalty}}$ penalizing the violations of the nonlinear constraints (2.13d) as $\mathcal{L}_{\mathrm{total}} = \mathcal{L}_{\mathrm{LS}} + \mathcal{L}_{\mathrm{penalty}}$

$$\mathcal{L}_{\mathrm{LS}} \triangleq \sum_{k=0}^{N-1} \frac{1}{2} \left(\left\| \boldsymbol{x}_k - \boldsymbol{x}_{\mathrm{r}_k} \right\|_{\boldsymbol{Q}}^2 + \left\| \boldsymbol{u}_k \right\|_{\boldsymbol{R}}^2 \right) + \frac{1}{2} \left\| \boldsymbol{x}_N - \boldsymbol{x}_{\mathrm{r}_N} \right\|_{\boldsymbol{Q}_N}^2 \tag{2.18a}$$

$$\mathcal{L}_{\mathrm{penalty}} \triangleq \sum_{k=0}^{N} \frac{1}{2} \begin{bmatrix} \boldsymbol{s}_{l_k} \\ \boldsymbol{s}_{u_k} \\ \mathbf{1} \end{bmatrix}^\top \begin{bmatrix} \boldsymbol{Q}_{s_l} & \mathbf{0} & \boldsymbol{p}_l \\ \mathbf{0} & \boldsymbol{Q}_{s_u} & \boldsymbol{p}_u \\ \boldsymbol{p}_l^\top & \boldsymbol{p}_u^\top & 0 \end{bmatrix} \begin{bmatrix} \boldsymbol{s}_{l_k} \\ \boldsymbol{s}_{u_k} \\ \mathbf{1} \end{bmatrix}. \tag{2.18b}$$

$\boldsymbol{Q} \in \mathbb{R}^{n \times n}$, and $\boldsymbol{R}^{m \times m}$ are the state and control running cost weight matrices respectively, while $\boldsymbol{Q}_N \in \mathbb{R}^{n \times n}$ is the terminal state cost weight. We assign both l_1 and l_2 penalties on the violations of the lower and upper bound nonlinear constraints (2.13d) associated with the slack variables $\boldsymbol{s}_{l_k}$ and $\boldsymbol{s}_{u_k}$ respectively, where $\boldsymbol{p}_l$, $\boldsymbol{p}_u$ are the l_1 penalty weights, and $\boldsymbol{Q}_{s_l}$, $\boldsymbol{Q}_{s_u}$ are the l_2 penalty weights. Notice that the slack variables are constrained to be positive in (2.13e) to attain the effect of an l_1 penalty as explained in [79].

Part II

Linear Stochastic Predictive Control

Chapter 3

Linear Stochastic MPC for Simplified Walking Model

This chapter builds upon this publication [34]. Linear Model Predictive Control (MPC) has been successfully used for generating feasible walking motions for humanoid robots. However, the effect of uncertainties on constraint satisfaction has only been studied using Robust MPC (RMPC) approaches, which account for the worst-case realization of bounded disturbances at each time instant. In this work, we propose for the first time to use linear stochastic MPC (SMPC) to account for uncertainties in bipedal walking. We show that SMPC offers more flexibility to the user (or a high-level decision-maker) by tolerating small (user-defined) probabilities of constraint violation. Therefore, SMPC can be tuned to achieve a constraint satisfaction probability that is arbitrarily close to 100%, without sacrificing performance as much as tube-based RMPC.

3.1 Tube-based Robust MPC (RMPC)

Two Tube-based linear RMPC versions were first introduced in [67] and [19]. We follow the approach of [67] as it has been more commonly used in the control community, and recently in [109] for bipedal locomotion. Note however that our qualitative results and comparison with SMPC would still hold for [19].

3.1.1 Robust OCP formulation and control objective

Consider the following discrete-LTI prediction model subject to additive stochastic disturbance w_t:

$$x_{t+i+1|t} = Ax_{t+i|t} + Bu_{t+i|t} + w_{t+i}, \tag{3.1a}$$
$$x_{t+i+1|t} \in \mathcal{X}, \tag{3.1b}$$
$$u_{t+i|t} \in \mathcal{U}. \tag{3.1c}$$

Assumption 1. *(Bounded disturbance) $w_{t+i} \in \mathcal{W}$ for $i = 0, 1, 2, ...$ is a disturbance realization, with $\mathcal{W}$ denoting a polytopic compact (closed and bounded) disturbance set containing the origin in its interior.*

Consider the nominal state s_t evolving as

$$s_{t+i+1|t} = As_{t+i|t} + Bv_{t+i|t}, \tag{3.2}$$

under the control action $v_{t+i|t}$. The main control objective of Tube-based RMPC is to bound the evolution of the closed-loop state error $e_t = x_t - s_t$ using an auxiliary state feedback control law

$$u_{t+i|t} = v_{t+i|t}(x_t) + K(x_{t+i|t} - s_{t+i|t}), \tag{3.3}$$

where $K \in \mathbb{R}^{m \times n}$ is a fixed pre-stabilizing feedback gain for (3.1a), and $v_{t+i|t}(s_t)$ is the decision variable of the MPC program. By subtracting (3.2) from (3.1a), and applying the control law in (3.3), the error dynamics is

$$e_{t+i+1} = A_K e_{t+i} + w_{t+i}, \tag{3.4}$$

with $A_K \triangleq A + BK$ being Schur (eigenvalues inside unit circle). The propagation of the closed-loop error dynamics (3.4) converges to the bounded set

$$\Omega = \bigoplus_{t=0}^{\infty} A_K^t \mathcal{W}. \tag{3.5}$$

Hence the limit set of all disturbed state trajectories x_t lies within a neighborhood of the nominal trajectory s_t known as a *tube of trajectories*. It is clear that if $\mathcal{W} = \{0\} \rightarrow \Omega = \{0\}$, and the tube of trajectories collapses to a single trajectory, which is the solution of (3.2). In set theory, Ω is called the *minimal Robust Positive Invariant* (mRPI) set, or *Infinite Reachable Set*. We recall some standard properties of disturbance invariant sets that will be used to design tightened constraint sets in the next subsection.

Property 1. *Positive Invariance*
A set $\mathcal{Z}$ is said to be a robust positively invariant (RPI) set [6] for the system (3.1a) iff

$$A_K \mathcal{Z} \oplus \mathcal{W} \subseteq \mathcal{Z}, \tag{3.6}$$

i.e. if $e_0 \in \mathcal{Z} \Rightarrow e_t \in \mathcal{Z} \ \forall t \geq 0$. In simple words, once the error is driven to $\mathcal{Z}$ it will remain inside $\mathcal{Z}$ for all future time steps if subject to the bounded disturbance $w_{t+i} \in \mathcal{W}$.

Property 2. *Minimal Robust Positive Invariance (mRPI)*
The mRPI set Ω (3.5) of (3.1a) is the RPI set in $\mathbb{R}^n$ that is contained in every closed RPI set of (3.1a).

An outer approximation of the mRPI set Ω can be computed using the approach of [87]. The size of Ω depends on the system's eigenvalues, the choice of K, and $\mathcal{W}$.

3.1.2 State and control back-off design

Using the mRPI set Ω, and the stabilizing feedback gains K, the state and control constraint sets are tightened as

$$s_{t+i+1|t} \in \mathcal{X} \ominus \Omega, \tag{3.7}$$
$$v_{t+i|t} \in \mathcal{U} \ominus K\Omega. \tag{3.8}$$

The new tightened state and control constraint sets are often called *backed-off constraints*. Satisfying the backed-off constraints (3.7)-(3.8) using the control law (3.3), ensures the satisfaction of (3.1b)-(3.1c).

Remark 1. *Following the choice of dead-beat pre-stabilizing feedback gains K proposed in [109], we get $K\Omega = K\mathcal{W}$, which allows us to compute $K\Omega$ exactly (whereas usually this needs to be approximated using numerical techniques). The dead-beat gains are also a practical choice since they lead to the smallest control back-off $K\Omega$ [109].*

3.1.3 Tube-based RMPC algorithm

The tube-based RMPC scheme solves the OCP in (3.1) by splitting it into two layers;

1. MPC layer: computes feasible feedfoward reference control actions $\mathbf{v}^*(s_t)$ every MPC sampling time t subject to the backed-off state and control

constraints as follows

$$\underset{\mathbf{v}}{\text{minimize}} \quad J_N(s_t, \mathbf{v}) = (2.7) \tag{3.9a}$$

subject to

$$s_{t+i+1|t} = As_{t+i|t} + Bv_{t+i|t}, \tag{3.9b}$$
$$s_{t+i+1|t} \in \mathcal{X} \ominus \Omega, \tag{3.9c}$$
$$v_{t+i|t} \in \mathcal{U} \ominus K\Omega, \tag{3.9d}$$
$$s_{t|t} = x_t, \tag{3.9e}$$
$$i = 0, 1, ..., N-1. \tag{3.9f}$$

2. State feedback control layer: employs the auxiliary state feedback control law (3.3) that regulates the feedforward term $v^*_{t|t}(s_t)$ such that the closed-loop error e_t is bounded inside Ω, which guarantees hard constraint satisfaction of (3.1b) - (3.1c).

Remark 2. *The above tube-based RMPC algorithm is often called closed-loop (CL) MPC since the initial state $s_{t|t} = x_t$ is the measured state x_t of the system [70][71][109]. However, due to disturbances, CL-MPC is not guaranteed to be recursively feasible (i.e. if the OCP is feasible at $t = 0$, it will remain feasible for all future time steps). One way to deal with recursive feasibility is to use $s_{t|t} = x_{t|t}$ whenever the OCP (3.9) is feasible, which is known as Mode 1. In case of infeasibility, we switch to a backup control strategy (Mode 2), where we use $s_{t|t} = s_{t+1|t-1}$, namely the current state from the previously optimized feasible trajectory [45]. In this case, recursive feasibility is guaranteed, and the resulting RMPC is not purely state-feedback, but a feedback controller comprising an extended state based on feasibility i.e. $u_{t+i|t} = v_{t+i|t}(x_t, s_{t+1|t-1}) + K(x_{t+i|t} - s_{t+i|t})$.*

3.2 Stochastic MPC With State And Control Chance Constraints (SMPC)

The main objectives of SMPC are to deal with computationally tractable stochastic uncertainty propagation for cost function evaluation, and to account for chance constraints, where constraints are expected to be satisfied within a desired probability level. With an abuse of notation, we will use some of the notations defined in Section 3.1 in a stochastic setting.

3.2.1 Stochastic (OCP) formulation and control objectives

Consider the following discrete-LTI prediction model subject to additive stochastic disturbance w_t:

$$x_{t+i+1|t} = Ax_{t+i|t} + Bu_{t+i|t} + w_{t+i}, \tag{3.10a}$$
$$\Pr[H_j x_{t+i+1|t} \leq h_j] \geq 1 - \beta_{x_j}, \quad j = 1, 2, ..., n_x \tag{3.10b}$$
$$\Pr[G_j u_{t+i|t} \leq g_j] \geq 1 - \beta_{u_j}, \quad j = 1, 2, ..., n_u \tag{3.10c}$$

Assumption 2. *(Stochastic disturbance)* $w_{t+i} \sim \mathcal{N}(0, \Sigma_w)$ *for* $i = 0, 1, 2, ...$ *is a disturbance realization of identically independent distributed (i.i.d.), zero mean random variables with normal distribution* $\mathcal{N}$*. The disturbance covariance* $\Sigma_w \in \mathbb{R}^{n \times n} = diag(\sigma_w^2)$[1] *is a diagonal matrix, with* $\sigma_w \in \mathbb{R}^n$*.*

Eq. (3.10b)/(3.10c) denote individual point-wise (i.e. independent at each point in time) linear state/control chance constraints with a maximum probability of constraint violation β_{x_j}/β_{u_j}. Since the disturbed state x_t in (3.10a) is now a stochastic variable, it is common to split its dynamics $x_{t+i|t} = s_{t+i|t} + e_{t+i|t}$ into two terms: a deterministic term $s_{t+i|t} = \mathbb{E}[x_{t+i|t}]$; and a zero-mean stochastic error term $e_{t+i|t} \sim \mathcal{N}(0, \Sigma_{x_{t+i|t}})$, which evolve as

$$s_{t+i+1|t} = As_{t+i|t} + Bv_{t+i|t}, \quad s_{t|t} = x_t \tag{3.11a}$$
$$e_{t+i+1|t} = A_K e_{t+i|t} + w_{t+i}, \quad e_{t|t} = 0. \tag{3.11b}$$

Notice that in contrast to the closed-loop error evolution in RMPC (3.4), the propagation of the predicted error $e_{t+i|t}$ in SMPC is independent of $x_{t+i|t}$ due to the assumption of zero initial error, which enables a closed-loop approach. The evolution of the state covariance

$$\Sigma_{x_{t+i+1|t}} = A_K \Sigma_{x_{t+i|t}} A_K^\top + \Sigma_w, \quad \Sigma_{x_{t|t}} = 0 \tag{3.12}$$

is independent of the control. In the same spirit as [59][40], the control objective is to bound the stochastic predicted error by employing the following control law:

$$u_{t+i|t} = v_{t+i|t}(x_t) + K(x_{t+i|t} - s_{t+i|t}). \tag{3.13}$$

$K \in \mathbb{R}^{m \times n}$ is a fixed stabilizing dead-beat feedback gains (see remark 1) for (3.10a), and $v_{t+i|t}$ is the decision variable of the MPC program. In what follows, we present a deterministic reformulation of the individual chance constraints (3.10b) - (3.10c) that will be used in the SMPC algorithm.

[1] $\sigma_w^2 \in \mathbb{R}^n = \left[\sigma_1^2, \sigma_2^2, ..., \sigma_n^2\right]^\top$ denotes the element-wise square operator of the standard deviation vector σ_w.

3.2.2 Chance constraints back-off design

Using the knowledge of the statistics of $x_{t+i|t}$ in (3.11a) - (3.11b), individual state chance constraints can be rewritten as:

$$\Pr[H_j s_{t+i+1|t} \leq h_j - H_j e_{t+i+1|t}] \geq 1 - \beta_{j_x}. \tag{3.14}$$

We seek the least conservative deterministic upper bound $\eta_{x_{j,t+i+1|t}}$ such that by imposing

$$H_j s_{t+i+1|t} \leq h_j - \eta_{x_{j,t+i+1|t}},$$

we can guarantee that (3.14) be satisfied. The smallest bound $\eta_{x_{j,t+i+1|t}}$ can then be obtained by solving $n_x N$ linear independent chance-constrained optimization problems offline:

$$\begin{aligned} \eta_{x_{j,t+i+1|t}} = \arg\min . \ & \eta_x \\ \text{subject to} \quad & \Pr[H_j e_{t+i+1|t} \leq \eta_x] \geq 1 - \beta_{x_j}. \end{aligned} \tag{3.15}$$

Using the disturbance assumption (2), one can solve such programs easily since there exist a numerical approximation of the cumulative density function (CDF) $\phi(\eta_{x_{j,t+i+1|t}}) \geq 1 - \beta_{x_j}$ for normal distribution. Hence, $\eta_{x_{j,t+i+1|t}}$ can be computed using the inverse of the CDF $\phi^{-1}(1-\beta_{x_j})$ of the random variable $H_j e_{t+i+1|t}$. Contrary to RMPC, the state back-offs grow contractively along the horizon, taking into account the predicted evolution of the error covariance. Similarly, we reformulate the individual control chance constraints in (3.10c) using (3.11a)-(3.11b), and the control law (3.13):

$$\Pr[G_j v_{t+i|t} \leq g_j - G_j K e_{t+i|t}] \geq 1 - \beta_{u_j}. \tag{3.16}$$

The individual control constraints back-off magnitudes $\eta_{u_{j,t+i|t}}$ can be computed along the horizon using the inverse CDF $\phi^{-1}(1 - \beta_{u_j})$ of the random variable $G_j K e_{t+i|t}$.

3.2.3 SMPC with chance constraints algorithm

The SMPC scheme with individual chance constraints computes feasible reference control actions $\mathbf{v}^*(x_t)$ at every MPC sampling time t subject to individual

state and control backed-off constraints as follows

$$\underset{\mathbf{v}}{\text{minimize}} \quad \mathbb{E}[J_N(x_t, \mathbf{v})] = (2.7) \tag{3.17a}$$

subject to

$$s_{t+i+1|t} = As_{t+i|t} + Bv_{t+i|t}, \tag{3.17b}$$

$$H_j s_{t+i+1|t} \leq h_j - \eta_{x_{t+i+1|t}},\ j = 0, 1, ..., n_x \tag{3.17c}$$

$$G_j v_{t+i|t} \leq g_j - \eta_{u_{t+i|t}}, \qquad j = 0, 1, ..., n_u \tag{3.17d}$$

$$s_{t|t} = x_t, \tag{3.17e}$$

$$i = 0, 1, ..., N-1. \tag{3.17f}$$

Note that since the above SMPC algorithm works purely with state-feedback ($s_{t|t} = x_t$), The linear feedback term in (3.13) is only used to predict the variance of the future error e_t.

Remark 3. *The above CL-SMPC algorithm is not guaranteed to be recursively feasible due to the fact that the disturbance realization $w_{t+i} \sim \mathcal{N}(0, \Sigma_w)$ is unbounded. To tackle this practically, disturbance realizations w_{t+i} are assumed to have a bounded support $\mathcal{W}$ [69]. There have been recent efforts on recursive feasibility for SMPC using different ingredients of cost functions, constraint tightening and terminal constraints as in [59] [84]. However, recursive feasibility guarantees for SMPC is out of this paper's scope.*

3.2.4 Worst-case Robustness of SMPC

SMPC ensures constraint satisfaction with a certain probability, while RMPC ensures it under bounded disturbances. When comparing the two approaches, one could think that SMPC is equivalent to bounding stochastic disturbances inside a confidence set and then applying RMPC. This section clarifies that this is not the case. In particular, we answer the following question: when using SMPC, what are the bounded disturbance sets under which we can still guarantee constraint satisfaction? Considering a single inequality constraint and hyper-rectangle disturbance sets, we show how to compute the size of such sets, and that they shrink along the control horizon. Since the disturbance set is instead fixed in RMPC, we conclude that the two approaches are fundamentally different.

Consider an individual chance constraint $\Pr[q^\top x_{t+i+1|t} \leq g] \geq 1 - \beta$, where $q \in \mathbb{R}^n$, $g \in \mathbb{R}$. Given the corresponding back-off magnitude $\eta_{t+i+1|t}$ (3.15), we seek the maximum hyper-rectangle disturbance set $\mathbb{W}_{t+i} \subset \mathbb{R}^n = \{w : |w| \leq w_{t+i}^{max}\}$

such that the constraint $q^\top x_{t+i+1|t} \leq g$ is satisfied for any $w \in \mathbb{W}_{t+i}$:

$$\eta_{t+i+1|t} = \max_{e} \; q^\top e \tag{3.18a}$$

$$\mathbf{s.t.} \quad e \in \bigoplus_{j=0}^{i} A_K^j \mathbb{W}_{t+i}.$$

This problem has a simple solution

$$\eta_{t+i+1|t} = \left(\sum_{j=0}^{i} \left| b_j \right|^\top \right) w_{t+i}^{\max}, \tag{3.19}$$

where $b_j^\top \triangleq q^\top A_K^j$ and $|.|$ is the element-wise absolute norm. From the SMPC derivation we know that $\eta_{t+i+1|t}$ is computed via the inverse CDF of $q^\top e_{t+i+1|t}$, which returns a value proportional to its standard deviation $\sigma_{t+i+1|t}$. Therefore we can write

$$\eta_{t+i+1|t} = \kappa(\beta) \underbrace{\sqrt{\sum_{j=0}^{i} b_j^\top \Sigma_w b_j}}_{\sigma_{t+i+1|t}}, \tag{3.20}$$

where $\kappa(\beta)$ is a coefficient that depends nonlinearly on β. By substituting (3.19) in (3.20) and exploiting the fact that $\Sigma_w = \text{diag}(\sigma_w^2)$ [2] $\in \mathbb{R}^{n \times n}$, we infer

$$\kappa^2(\beta) \sum_{j=0}^{i} b_j^\top \, \text{diag}(b_j) \, \sigma_w^2 = (\sum_{j=0}^{i} |b_j|^\top w_{t+i}^{\max})^2. \tag{3.21}$$

Solving for $w_{t+i}^{\max}$ has infinitely many solutions. However, we can get a unique solution by imposing a ratio $\zeta_{t+i} \in \mathbb{R}$ between $w_{t+i}^{\max}$ and σ_w as follows:

$$w_{t+i}^{\max} = \zeta_{t+i} \, \sigma_w. \tag{3.22}$$

Substituting back in (3.21) and solving for ζ_{t+i} we get:

$$\zeta_{t+i} = \kappa(\beta)\sqrt{\alpha_i}, \qquad \alpha_i \triangleq \frac{\sum_{j=0}^{i} b_j^\top \, \text{diag}(b_j) \, \sigma_w^2}{(\sum_{j=0}^{i} |b_j|^\top \sigma_w)^2}. \tag{3.23}$$

[2] $\sigma_w^2 \in \mathbb{R}^n = \left[\sigma_1^2, \sigma_2^2, ..., \sigma_n^2\right]^\top$ denotes the element-wise square operator of the standard deviation vector σ_w.

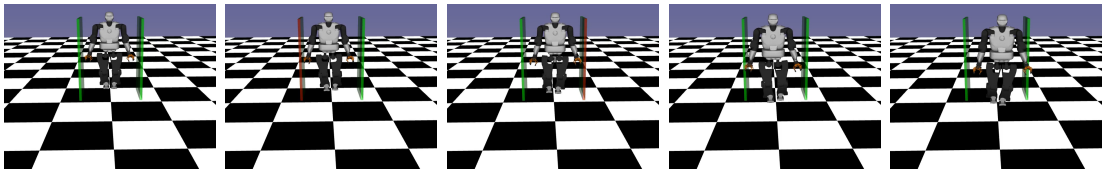


Figure 3.1: TALOS robot walking through a narrow hallway using nominal MPC subject to additive disturbances on the lateral CoM dynamics. The Red color corresponds to the robot colliding with the wall.

The series α_i is bounded $0 < \alpha_i \leq 1, \quad \forall i \geq 0$, since the sum of squares (numerator) is less than or equal to the square of the sum of positive numbers (denominator). In Appendix 3.4, we prove that α_i is monotonically decreasing (i.e. $\alpha_{i+1} \leq \alpha_i$) for the case of 1D systems ($n = 1$). We confirmed this result numerically for the multi-variate case by randomly generating schur stable closed-loop matrices $A + BK$ subject to the same covariance of the disturbance Σ_w for fairness. Since α_i is bounded and monotonically decreasing, then it is convergent. This implies that, as i grows, ζ_{t+i} decreases, and so does the disturbance set $\mathbb{W}_{t+i}$ until it converges in the limit. We conclude that, when using SMPC, the disturbance sets for which we have guaranteed constraint satisfaction shrink along the control horizon.

3.3 Simulation Results

In this section, we present simulation results of the generated walking motions of a Talos robot [100] subject to additive persistent disturbances on the lateral CoM dynamics. We compare the motions generated using SMPC subject to state and control chance constraints against nominal MPC and tube-based RMPC. The lateral CoM position is constrained inside a box $-0.04 \leq c^y \leq 0.04$ to avoid collision of the external parts of the robot with walls as it navigates through a narrow hallway with fixed contact locations as shown in Fig. 3.1. The CoM trajectories generated using MPC are tracked with a Task-Space Inverse Dynamics (TSID) controller using a hard contact model for generating the control commands [26]. We use the Pinocchio library [16] for the computation of rigid-body dynamics. We show an empirical study comparing robustness w.r.t. performance of SMPC against tube-based RMPC and nominal MPC when subject to the same disturbance realizations. The robot model and simulation parameters are defined in Table 3.1.

3.3.1 Hard constraints satisfaction in tube-based RMPC

First, we compute offline the state and control back-off magnitudes to tighten the constraint sets for RMPC. The state constraints back-off magnitude is computed

Table 3.1: Modelling and simulation parameters.

CoM height (h)	0.88 (m)
gravity acceleration (g^z)	9.81 (m/s^2)
foot support polygon along x direction ($\mathcal{U}^x$)	$[-0.05, 0.10]$ (m)
foot support polygon along y direction ($\mathcal{U}^y$)	$[-0.05, 0.05]$ (m)
bounded disturbance on CoM position ($\mathcal{W}_c$)	$[-0.0016, 0.0016]$ (m)
bounded disturbance on CoM velocity ($\mathcal{W}_{\dot{c}}$)	$[-0.016, 0.016]$ (m/s)
disturbance std-dev of CoM position (σ_c)	0.0008 (m)
disturbance std-dev of CoM velocity ($\sigma_{\dot{c}}$)	0.008 (m/s)
MPC sampling time (Δt)	0.1 (s)
whole-body tracking controller sampling time	0.002 (s)
MPC receding horizon (N)	16

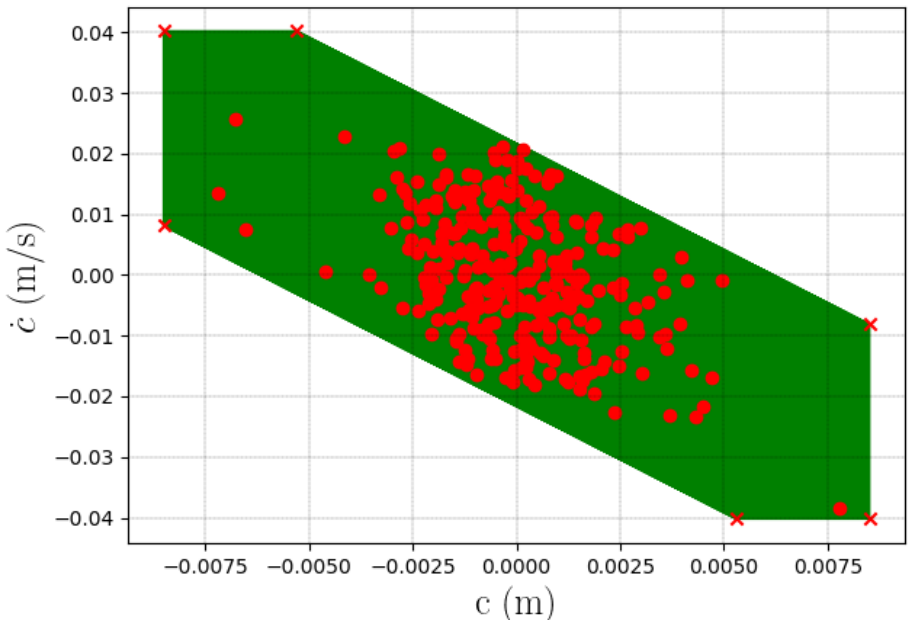


Figure 3.2: Simulation of 6 initial conditions (red crosses) at the vertices of the outer-ϵ approximation of the mRPI set Ω for 50 time steps subject to $w_{t+i} \in \mathcal{W}$.

using an outer ϵ approximation of the mRPI set Ω using the procedure in [87], with an accuracy of $\epsilon = 10^{-6}$. In Fig. 3.2, we test the positive invariance property (1) of Ω, by simulating 6 initial conditions starting at the set vertices for 50 time steps, and applying randomly sampled disturbance realizations from the disturbance set $\mathcal{W}$. As shown, the evolution of each initial condition (red dots), is kept inside Ω (the tube section) when subject to disturbance realizations $w_{t+i} \in \mathcal{W} = \begin{bmatrix} \mathcal{W}_c & \mathcal{W}_{\dot{c}} \end{bmatrix}^\top$. Using the same choice of pre-stabilizing dead-beat gains $K = \begin{bmatrix} 3.386 & 0.968 \end{bmatrix}$ as in [109], the robust control back-off magnitude $K\Omega$ is computed exactly without resorting to numerical approximation $K\Omega = K\mathcal{W} = \begin{bmatrix} -0.0225, & 0.0225 \end{bmatrix}$.

In Fig. 3.3, we plot the CoM position and CoP of 200 trajectories obtained using tube-based RMPC. The robot takes the first two steps in place before entering the hallway. In the third and fourth steps, no disturbances were applied showing that the CoM position c trajectories back off conservatively from the constraint bounds with the magnitude of the mRPI set on the CoM position Ω_c. Finally, we randomly apply sampled Gaussian disturbance realizations $w_{t+i} \sim \mathcal{N}(0, \Sigma_w)$ with finite support $\mathcal{W}$, where $\Sigma_w = \begin{bmatrix} \sigma_c^2 & 0 \\ 0 & \sigma_{\dot{c}}^2 \end{bmatrix}$, for the rest of the motion, showing that both state and control constraints are satisfied as expected. Note that when the worst-case disturbance is persistently applied on one direction, the state constraint is saturated in that direction as expected. This shows that tube-based RMPC anticipates for a persistent worst-case disturbance to guarantee a hard constraint satisfaction, which is quite conservative and sub-optimal when the nature of the disturbances is stochastic as in this scenario.

3.3.2 Chance-constraints satisfaction in SMPC

This subsection presents the results of SMPC. Contrary to RMPC, the state and control back-off magnitudes ($\eta_{x_{t+i+1|t}}$, $\eta_{u_{t+i}}$) vary along the horizon, and are computed based on the propagation of the predicted state covariance (3.12), pre-stabilizing feedback gain K, disturbance covariance Σ_w, and the desired probability level of individual state and control constraint violation β_{x_j} and β_{u_j} respectively. We set $\beta_{x_j} = 5\%$, and $\beta_{u_j} = 50\%$, which corresponds to satisfying the nominal CoP constraints.

Using the same choice of stabilizing feedback gains K as in RMPC, we simulate 200 trajectories using SMPC in Fig. 3.4b. In the first two steps, the robot steps in place and the CoM constraints are not active. For the rest of the motion, we randomly apply sampled Gaussian disturbance realizations $w_{t+i} \sim \mathcal{N}(0, \Sigma_w)$ with finite support $\mathcal{W}$. In Fig. 3.4a, we show the empirical number of CoM position constraint violations out of the 200 simulated trajectories. The maximum number of constraint violations is obtained at time instance 4.3s is $5 (\leq 10)$, which respects

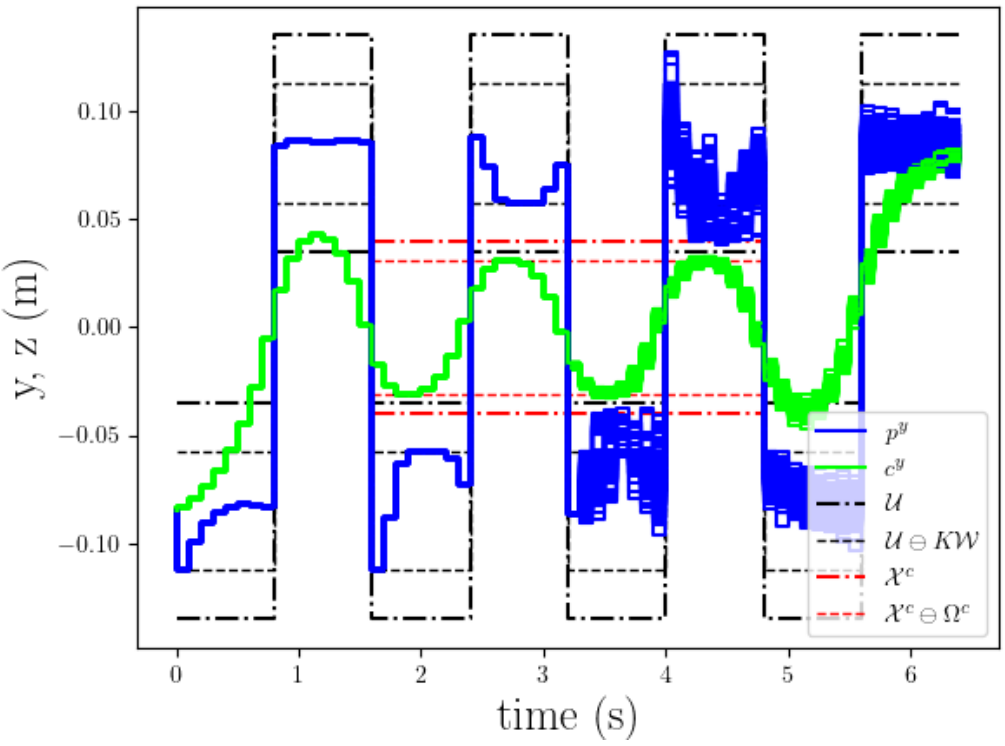
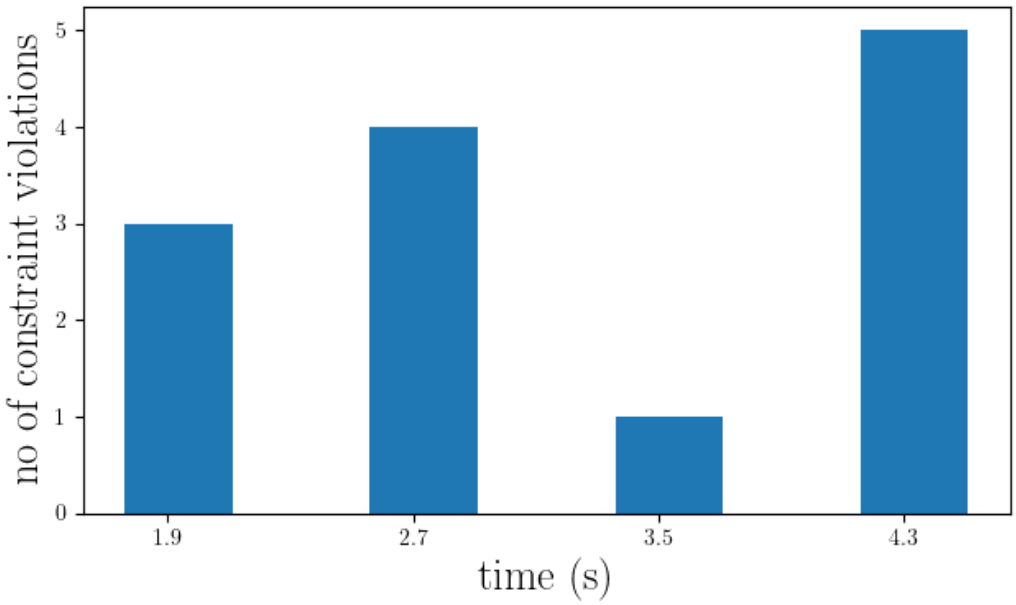
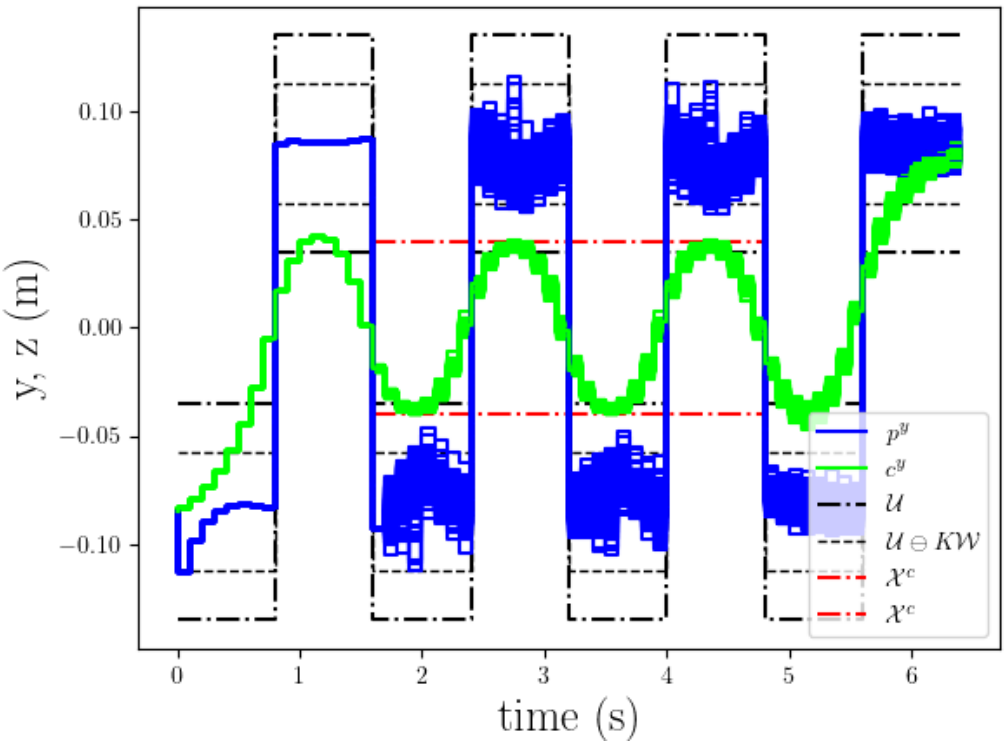


Figure 3.3: 200 simulations of tube-based RMPC with $w_{t+i} \in \mathcal{W}$.



(a) CoM position constraint violations.



(b) CoP and CoM lateral motion with $\beta_{x_j} = 5\%$, $\beta_{u_j} = 50\%$.

Figure 3.4: 200 SMPC simulations with $w_{t+i} \sim \mathcal{N}(0, \Sigma_w) \in \mathcal{W}$.

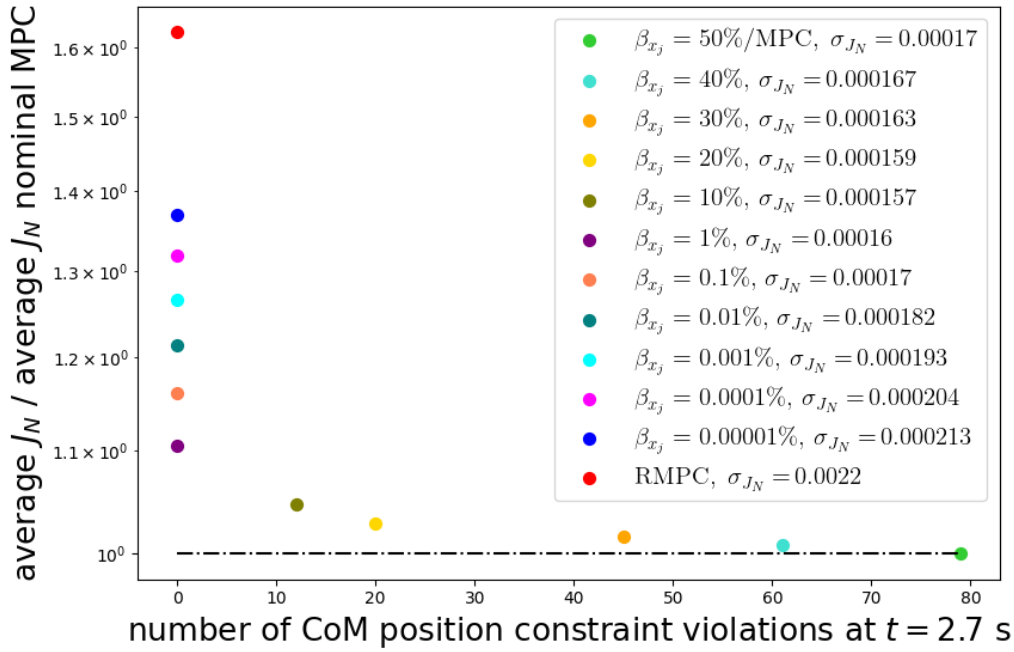


Figure 3.5: SMPC with varying β_{x_j} vs RMPC of 200 simulations with $w_{t+i} \sim \mathcal{N}(0, \Sigma_w) \in \mathcal{W}$. The dotted line denotes the optimal ratio of one (nominal MPC)

the designed probability level of CoM constraint violations $\beta_{x_j} = 5\%$ as expected. To test robustness of constraint satisfaction and optimality of SMPC, we ran an empirical study of the same eight step walking motion (200 trajectories) comparing SMPC with varying $\beta_{x_j} \in \begin{bmatrix} 0.00001\%, & 50\% \end{bmatrix}$ and fixed $\beta_{u_j} = 50\%$ against tube-based RMPC and nominal MPC in Fig. 3.5. We plot the empirical number of CoM position constraint violations at $t = 2.7$s, against the averaged cost performance (of 200 trajectories) ratio between different MPC schemes and nominal MPC. As before, disturbance realizations are sampled from $\mathcal{N}(0, \Sigma_w)$ with finite support $\mathcal{W}$. As expected, the higher the probability level of constraint satisfaction in SMPC, the lower the amount of constraint violations (higher robustness). The highest number of constraint violations is obtained at $\beta_{x_j} = 50\%$, which is equivalent to nominal MPC. Zero constraint violations were obtained when $\beta_{x_j} \leq 1\%$, as for RMPC. An advantage of SMPC with $\beta_{x_j} \leq 1\%$ over RMPC, is the lower average cost. This gives the user the flexibility to design the controller for different task constraints, by tuning the probability level of constraint satisfaction without sacrificing performance as in tube-based RMPC or sacrificing robustness as in nominal MPC.

3.4 Conclusions

This work compared the use of SMPC with RMPC to account for uncertainties in bipedal locomotion. Many SMPC and RMPC algorithms exist. We decided to focus on two particular instances of tube-based approaches, which have the same online computational complexity as nominal MPC. Indeed, all the extra computation takes place offline, and consists in the design of tightened constraints (back-offs) based on a fixed pre-stabilizing feedback gain K. Our comparison focused on the trade off between robustness and optimality. Our tests show that, while SMPC does not provide hard guarantees on constraint satisfaction, in practice we did not observe any constraint violation with sufficiently low $\beta_x (\leq 1\%)$. This comes with the advantage of less conservative control, i.e. it results in better performance as measured by the cost function. This is reasonable because RMPC behaves conservatively, expecting a persistent worst-case disturbance, which in practice is extremely unlikely to happen. SMPC instead reasons about the probability of disturbances. In Section (3.2.4) we showed that we can compute the maximum disturbance sets to which SMPC ensures robustness. We showed that these sets shrink contractively as time grows. Loosely speaking, SMPC can be thought as a special kind of RMPC that considers shrinking disturbance sets along the horizon.

Our empirical results are specific to the choice of dead-beat feedback gains used in both algorithms. These gains were computed in [109] by minimizing the back-off magnitude on the CoP constraints. This is sensible because the CoP is usually more constrained than the CoM in bipedal locomotion. Other feedback gains could be used, such as LQR gains, resulting in back-off magnitudes that are a trade-off between state and control constraints. While changing the gains would affect our quantitative results, it would not affect the qualitative differences between SMPC and RMPC that we highlighted in the paper. In conclusion, SMPC offers an opportunity for the control of walking robots that affords trading-off robustness to uncertainty and performance. For Future work, we intend to investigate nonlinear versions of RMPC and SMPC [56],[91] to enable the use of more complex models of locomotion.

3.5 Appendix

Proof that α_i is monotonically decreasing (1D case):
We would like to show that $\alpha_{i+1} \leq \alpha_i \,,\ \forall i \geq 0$. Given

$$\alpha_i = \frac{\sum_{j=0}^{i} b_j^2 \, \sigma_w^2}{(\sum_{j=0}^{i} |b_j| \sigma_w)^2} \quad \alpha_{i+1} = \frac{\sum_{j=0}^{i+1} b_j^2 \, \sigma_w^2}{(\sum_{j=0}^{i+1} |b_j| \sigma_w)^2}, \tag{3.24}$$

where $b_j = q^\top A_K^j$ can be written as qa^j, with $a \triangleq A_K$, $|a| < 1$. After simplifying σ_w, then $\alpha_{i+1} \leq \alpha_i$ reads as:

$$\frac{\sum_{j=0}^{i} b_j^2 + b_{i+1}^2}{(\sum_{j=0}^{i} |b_j|)^2 + b_{i+1}^2 + 2|b_{i+1}| \sum_{j=0}^{i} |b_j|} \leq \frac{\sum_{j=0}^{i} b_j^2}{(\sum_{j=0}^{i} |b_j|)^2}. \tag{3.25}$$

By substituting the analytical expressions of the following series in (3.25)

$$\sum_{j=0}^{i} |b_j| = |q| \sum_{j=0}^{i} |a|^j = |q| \left(\frac{1 - |a|^{i+1}}{1 - |a|} \right), \tag{3.26a}$$

$$\sum_{j=0}^{i} b_j^2 = q^2 \sum_{j=0}^{i} a^{2j} = q^2 \left(\frac{1 - a^{2(i+1)}}{1 - a^2} \right). \tag{3.26b}$$

and cross multiplication, we get

$$|q|^3 \left(\frac{1 - |a|^{i+1}}{1 - |a|} \right)^2 |a|^{i+1} \leq q^2 \left(\frac{1 - a^{2(i+1)}}{1 - a^2} \right) \left(|qa^{i+1}| + 2|q| \left(\frac{1 - |a|^{i+1}}{1 - |a|} \right) \right). \tag{3.27}$$

By multiplying both sides of (3.27) by $\frac{(1-|a|)^2}{|q|^3}$, we have

$$\begin{aligned} |a|^{i+1}(1 - |a|^{i+1})^2 &\leq \frac{1 - a^{2(i+1)}}{1 + a} \left(|a|^{i+1}(1 - |a|) + 2(1 - |a|^{i+1}) \right) \\ \Rightarrow (1 + |a|)(|a|^{i+1} - a^{2i+2}) &\leq (1 + |a|^{i+1})(-|a|^{i+1} - |a|^{i+2} + 2) \\ \Rightarrow |a|^{i+1} + |a|^{i+2} - |a|^{2i+3} &\leq 2 + |a|^{i+1} - |a|^{i+2} - |a|^{2i+3} \\ \Rightarrow 2 - 2|a|^{i+2} &\geq 0, \end{aligned} \tag{3.28}$$

which always holds because $|a| < 1$. This concludes the proof. ■

Part III

Nonlinear Stochastic Predictive Control

Chapter 4

Nonlinear Stochastic Trajectory Optimization for Centroidal Momentum Motion Generation

This chapter builds upon this publication [33]. Generation of robust trajectories for legged robots remains a challenging task due to the underlying nonlinear, hybrid, and intrinsically unstable dynamics which need to be stabilized through limited contact forces. Furthermore, disturbances arising from unmodelled contact interactions with the environment and model mismatches can hinder the quality of the planned trajectories leading to unsafe motions. In this work, we propose to use stochastic trajectory optimization for generating robust centroidal momentum trajectories to account for additive uncertainties in the model dynamics and parametric uncertainties on contact locations. Through an alternation between the robust centroidal and whole-body trajectory optimizations, we generate robust momentum trajectories while being consistent with the whole-body dynamics. We perform an extensive set of simulations subject to different uncertainties on a quadruped robot showing that our stochastic trajectory optimization problem reduces the amount of foot slippage for different gaits while achieving better performance over deterministic planning.

4.1 Stochastic Optimal Control for Centroidal Momentum Trajectory Optimization

In this section, we present a stochastic version of the problem (1) that takes into account additive stochastic uncertainties on the centroidal momentum dynamics as well as contact position uncertainties subject to friction pyramid chance constraints. We consider the following discrete-time stochastic nonlinear OCP:

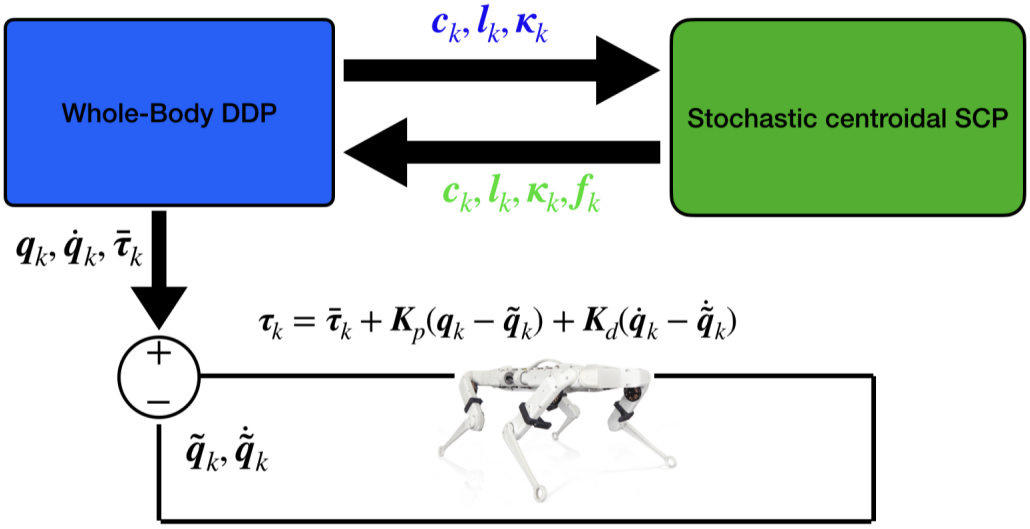


Figure 4.1: Robust trajectory optimization framework alternating between centroidal states of whole-body DDP motions and stochastic centroidal SCP motions.

Problem 3. *Stochastic Optimal Control Problem (SOCP)*

$$\underset{\boldsymbol{x},\boldsymbol{u}}{\text{minimize}} \quad l_f(\boldsymbol{x}_N) + \sum_{i=0}^{N-1} l(\boldsymbol{x}_k, \boldsymbol{u}_k) \tag{4.1a}$$

subject to

$$\boldsymbol{x}_{k+1} = \boldsymbol{f}(\boldsymbol{x}_k, \boldsymbol{u}_k, \boldsymbol{\theta}_k, \boldsymbol{w}_k), \tag{4.1b}$$
$$Pr(\boldsymbol{H}\boldsymbol{x}_{\boldsymbol{k}} \leq \boldsymbol{h}) \geq \alpha_x, \tag{4.1c}$$
$$Pr(\boldsymbol{G}\boldsymbol{u}_k \leq \boldsymbol{g}) \geq \alpha_u, \tag{4.1d}$$
$$\boldsymbol{x}_0 = \boldsymbol{x}(0), \tag{4.1e}$$
$$\boldsymbol{x}_f = \boldsymbol{x}(N), \tag{4.1f}$$
$$\forall k \in \{0, 1, .., N-1\}. \tag{4.1g}$$

With abuse of notation from Problem (1), $\boldsymbol{x}_k$ and $\boldsymbol{u}_k$ will be considered the stochastic state and control policies evolving according to the parametric and additive stochastic disturbance realizations $\boldsymbol{\theta}_k$, and $\boldsymbol{w}_k$. (4.1c)-(4.1d) are the state and control polytopic joint chance constraints with α_x and α_u being the probability levels of state and control constraint satisfaction respectively.

Assumption 3. *(i.i.d. Gaussian disturbances)*
$\boldsymbol{\theta}_k \sim \mathcal{N}(\mathbb{E}[\boldsymbol{\theta}_k], \boldsymbol{\Sigma}_{\boldsymbol{\theta}_k})$, *and* $\boldsymbol{w}_k \sim \mathcal{N}(\mathbb{E}[\boldsymbol{w}_k], \boldsymbol{\Sigma}_{\boldsymbol{w}_k})$ *are assumed to be independent*

and identically distributed (i.i.d.) disturbance realisations following Gaussian distributions. $\mathbb{E}[\boldsymbol{\theta}_k] = \boldsymbol{p}_{e,i_k}$, *and* $\boldsymbol{\Sigma}_{\boldsymbol{\theta}_k} \in \mathbb{R}^{(3n_c \times 3n_c)}$ *represent the mean and covariance of the contact positions respectively.* $\mathbb{E}[\boldsymbol{w}_k] = \mathbf{0}$ *and* $\boldsymbol{\Sigma}_{\boldsymbol{w}_k} \in \mathbb{R}^{9\times 9}$ *are the mean and covariance of the additive noise on the centroidal dynamics.*

4.1.1 Individual Chance Constraints Reformulation

Solving the above joint chance constraints (4.1c)-(4.1d) involves the integration of multi-dimensional Gaussian Probability Density Functions (PDFs), which becomes computationally intractable for high dimensions. One effective solution is to use Boole's inequality:

$$\Pr(\bigvee_{i=1}^{n} \boldsymbol{C}_i) \leq \sum_{i}^{n} \Pr(\boldsymbol{C}_i) \tag{4.2}$$

as a conservative union bound on the joint chance constraints [80]. We can rewrite the complement of the state chance constraints as a conjunction of individual chance constraints as follows:

$$(4.1c) = \Pr(\bigwedge_{i=1}^{l_x} \boldsymbol{H}_i \boldsymbol{x} \leq h_i) \geq \alpha_x,$$

which can be written conservatively as

$$\Pr(\bigvee_{i=1}^{l_x} \boldsymbol{H}_i \boldsymbol{x} > h_i) \leq 1 - \alpha_x. \tag{4.3}$$

By applying Boole's inequality on the above equation, and allocating constraint violation risk equally $\epsilon_{x_i} = (1-\alpha_x)/l_x$, with l_x being the number of intersecting hyper-planes forming the state joint polytopic constraint, we reach

$$\begin{aligned}(4.3) \xleftarrow{(4.2)} \sum_{i=1}^{l_x} \Pr(\boldsymbol{H}_i \boldsymbol{x} > h_i) &\leq \epsilon_{x_i} \\ \equiv \sum_{i=1}^{l_x} \Pr(\boldsymbol{H}_i \boldsymbol{x} \leq h_i) &\geq 1 - \epsilon_{x_i}. \end{aligned} \tag{4.4}$$

Similarly, control joint chance constraint (4.1d) can be reformulated as a set of individual chance constraints following the same arguments as before

$$\sum_{i=1}^{l_u} \Pr(\boldsymbol{G}_i \boldsymbol{u} \leq g_i) \geq 1 - \epsilon_{u_i}, \tag{4.5}$$

where $\epsilon_{u_i} = (1-\alpha_u)/l_u$ is the equally distributed control constraint risk.

Remark 4. *Allocating the risk of constraint violations equally can be quite conservative since one would preferably allocate more risk to active constraints over inactive ones. Another approach can be optimized for allowable violation for each constraint as in [60], which involves higher computational complexity.*

4.1.2 Deterministic Reformulation of Individual Chance Constraints

Solving the chance constraints (4.4)-(4.5), requires propagating the uncertainty through the nonlinear dynamics. We adopt a linearization-based covariance propagation as in [121][57]. Using a state-feedback control policy $\boldsymbol{u}_k = \boldsymbol{v}_k + \boldsymbol{K}_k(\boldsymbol{x}_k - \boldsymbol{s}_k)$, where $\boldsymbol{K}_k$ are pre-stabilizing feedback gains, then the mean and covariance of the dynamics evolve as

$$\boldsymbol{s}_{k+1} \approx \bar{\boldsymbol{f}}(\boldsymbol{s}_k, \boldsymbol{v}_k, \boldsymbol{p}_{e,k}, \boldsymbol{0}) + \boldsymbol{A}_k(\boldsymbol{s}_k - \boldsymbol{s}_k^j) + \boldsymbol{B}_k(\boldsymbol{v}_k - \boldsymbol{v}_k^j), \tag{4.6a}$$

$$\boldsymbol{\Sigma}_{\boldsymbol{x}_{k+1}} = \boldsymbol{A}_{\text{cl}}\boldsymbol{\Sigma}_{\boldsymbol{x}_k}\boldsymbol{A}_{\text{cl}}^\top + \boldsymbol{C}_k\boldsymbol{\Sigma}_{\boldsymbol{\theta}}\boldsymbol{C}_k^\top + \boldsymbol{\Sigma}_{\boldsymbol{w}}, \tag{4.6b}$$

where $\bar{\boldsymbol{f}}$ is the nominal nonlinear dynamics estimated at the current mean of the state $\boldsymbol{s}_k^j$ and controls $\boldsymbol{v}_k^j$ of the jth trajectory. $\boldsymbol{\Sigma}_{\boldsymbol{x}_0} = \boldsymbol{0}$, and $\boldsymbol{A}_{\text{cl}} \triangleq \boldsymbol{A}_k + \boldsymbol{B}_k\boldsymbol{K}_k$ is the closed-loop dynamics. $\boldsymbol{A}_k \triangleq \frac{\partial}{\partial \boldsymbol{s}}\boldsymbol{f}(\boldsymbol{s}_k, \boldsymbol{v}_k, \boldsymbol{p}_{e,k}, \boldsymbol{0})|_{(\boldsymbol{s}_k^j, \boldsymbol{v}_k^j)}$ is the Jacobian of the dynamics w.r.t. the state. $\boldsymbol{B}_k \triangleq \frac{\partial}{\partial \boldsymbol{v}}\boldsymbol{f}(\boldsymbol{s}_k, \boldsymbol{v}_k, \boldsymbol{p}_{e,k}, \boldsymbol{0})|_{(\boldsymbol{s}_k^j, \boldsymbol{v}_k^j)}$ is the Jacobian of the dynamics w.r.t. controls. Finally, $\boldsymbol{C}_k \triangleq \frac{\partial}{\partial \boldsymbol{p}_{e,k}}\boldsymbol{f}(\boldsymbol{s}_k, \boldsymbol{v}_k, \boldsymbol{p}_{e,k}, \boldsymbol{0})|_{(\boldsymbol{s}_k^j, \boldsymbol{v}_k^j)}$ represents the Jacobian of the dynamics w.r.t. the contact positions.

Remark 5. *Other approaches can be used for uncertainty propagation through nonlinear dynamics like unscented-based transforms [85], or Generalized Polynomial Chaos (gPC) [77]. These methods can lead to a more accurate estimate of the propagated uncertainty at the cost of a significant increase in complexity. Since computational efficiency is more important in our case (especially for online re-planning of the trajectories), we prefer to not use these methods.*

Based on Assumption (3) and the covariance propagation in (4.6b), we seek the least conservative upper bounds on the state and control individual chance-constraints (4.1c)-(4.1d). Using the inverse of the *Cumulative Density Function* (CDF) ϕ^{-1} of a Gaussian distribution, we arrive at a deterministic reformulation of the chance constraints:

$$\boldsymbol{H}_i\boldsymbol{s}_k \leq h_i - \eta_{x_{i,k}}, \tag{4.7a}$$

$$\boldsymbol{G}_i\boldsymbol{v}_k \leq g_i - \eta_{u_{i,k}}, \tag{4.7b}$$

where $\eta_{x_{i,k}} = \phi^{-1}(1-\epsilon_{x_i})\|\boldsymbol{H}_i\|_{\boldsymbol{\Sigma}_k}$ and $\eta_{u_{i,k}} = \phi^{-1}(1-\epsilon_{u_i})\|\boldsymbol{G}_i\boldsymbol{K}_k\|_{\boldsymbol{\Sigma}_k}$ are known as the state and control back-off bounds ensuring the satisfaction of the individual chance constraints (4.1c)-(4.1d), respectively.

4.1.3 Deterministic Reformulation of SOCP

Given the previous reformulation of the individual chance constraints, we can write down the following NOCP.

Problem 4. *NOCP with reformulated individual chance-constraints:*

$$\underset{\boldsymbol{s},\boldsymbol{v}}{\text{minimize}} \quad l_f(\boldsymbol{s}_N) + \sum_{i=0}^{N-1} l(\boldsymbol{s}_k, \boldsymbol{v}_k) \tag{4.8a}$$

subject to

$$\boldsymbol{s}_{k+1} = \boldsymbol{f}(\boldsymbol{s}_k, \boldsymbol{v}_k, \boldsymbol{p}_{e,k}, \boldsymbol{0}), \tag{4.8b}$$

$$\boldsymbol{H}_{i,k}\boldsymbol{s}_k \leq h_{i,k} - \eta_{x_{i,k}} \qquad \forall i \in \{1, 2, .., l_x\}, \tag{4.8c}$$

$$\boldsymbol{G}_{i,k}\boldsymbol{v}_k \leq g_{i,k} - \eta_{u_{i,k}} \qquad \forall i \in \{1, 2, .., l_u\}, \tag{4.8d}$$

$$\boldsymbol{s}_0 = \boldsymbol{s}(0), \tag{4.8e}$$

$$\boldsymbol{s}_f = \boldsymbol{s}(N), \tag{4.8f}$$

$$\forall k \in \{0, 1, .., N-1\}. \tag{4.8g}$$

where (4.8b) is now the mean of the nonlinear dynamics. To solve the above nonlinear OCP, we resort to Sequential Convex Programming (SCP), which we explain in the next subsection.

4.1.4 SCP with L1 Trust Region Penalty Cost

SCP attempts to solve nonlinear OCPs by successively linearizing the dynamics, costs, and constraints to solve a convex sub-problem at every iteration. The dynamics are linearized with a first-order Taylor expansion around the previous state and control trajectories computed at the j-th succession. Successive linearization introduces two well-known problems [60].

1) Artificial infeasibility: the problem becomes infeasible even if the original nonlinear problem is feasible. The most evident example of this arises when the problem is linearized about an unrealistically short time horizon so that there is no feasible control input that can satisfy the prescribed dynamics and constraints. 2) Artificial unboundedness: the solution takes steps far away from the validity of the linear model. To mitigate artificial unboundedness, a trust-region constraint is employed. Different approaches are adapted to tackle artificial infeasibility. In [60], the authors employ hard constraints and virtual controls as slack variables on the constraints. However, [8] enforced hard constraints on the dynamics and convex soft penalties on the rest of the constraints along with trust region constraints. In this work we follow the same rationale as [8, 95], where the trust region constraints

$c_i(\boldsymbol{x}) \leq 0$ are enforced as l_1 penalty cost in the form of

$$\arg\min .\{\arg\max .\gamma(c_i(\boldsymbol{x}), 0)\}, \tag{4.9a}$$

$$c_i(\boldsymbol{x}) = |\boldsymbol{x}_k - \boldsymbol{x}_k^j| - \Omega. \tag{4.9b}$$

where Ω is the trust region radius. Notice that the above l_1 penalty cost is exact—meaning that as the penalty weight γ gets infinitely large, the constraint violations are driven to zero. Even though (4.9) is non-differentiable, it can be solved efficiently by introducing a slack variable t as follows:

$$\underset{t}{\text{minimize}} \quad \gamma t \tag{4.10a}$$

$$\text{subject to}$$

$$|\boldsymbol{x} - \boldsymbol{x}^j| - \Omega \leq t, \tag{4.10b}$$

$$-t \leq 0. \tag{4.10c}$$

To solve problem (4), we solve a sequence of Quadratic Programs (QPs) in problem (5), accompanied by a trust region update mechanism based on the accuracy ratio of the linearized model w.r.t. the nonlinear model as in [8][57].

Problem 5. *Convexified QP at the j-th SCP iteration:*

$$\underset{\boldsymbol{s},\boldsymbol{v},\boldsymbol{t}}{\text{minimize}} \quad l_f(\boldsymbol{s}_N) + \sum_{i=0}^{N-1} l(\boldsymbol{s}_k, \boldsymbol{v}_k) + \gamma^j \sum_{i=0}^{N} t_k \tag{4.11a}$$

subkect to

$$\boldsymbol{s}_{k+1} = \bar{\boldsymbol{f}}(\boldsymbol{s}_k, \boldsymbol{v}_k, \boldsymbol{p}_{e,k}, \boldsymbol{0}) + \boldsymbol{A}_k(\boldsymbol{s}_k - \boldsymbol{s}_k^j) + \boldsymbol{B}_k(\boldsymbol{v}_k - \boldsymbol{v}_k^j), \tag{4.11b}$$

$$\boldsymbol{\Sigma}_{k+1} = \boldsymbol{A}_{cl}\boldsymbol{\Sigma}_{\boldsymbol{x}_k}\boldsymbol{A}_{cl}^\top + \boldsymbol{C}_k\boldsymbol{\Sigma}_{\boldsymbol{\theta}}\boldsymbol{C}_k^\top + \boldsymbol{\Sigma}_{\boldsymbol{w}}, \tag{4.11c}$$

$$\boldsymbol{\Sigma}_0 = \boldsymbol{0}_{9\times 9}, \tag{4.11d}$$

$$\boldsymbol{H}_{i,k}\boldsymbol{s}_k \leq h_{i,k} - \phi^{-1}(1-\epsilon_{x_i})\Big(\big\|\boldsymbol{H}_{i,k}\big\|_{\boldsymbol{\Sigma}_k} + \frac{\partial}{\partial \boldsymbol{z}}\big\|\boldsymbol{H}_{i,k}\big\|_{\boldsymbol{\Sigma}_k}(\boldsymbol{z}_k - \boldsymbol{z}_k^j)\Big), \quad \forall i \in \{1,2,..,l_x\}, \tag{4.11e}$$

$$\boldsymbol{G}_{i,k}\boldsymbol{v}_k \leq g_{i,k} - \phi^{-1}(1-\epsilon_{u_i})\Big(\big\|\boldsymbol{G}_{i,k}\boldsymbol{K}_k\big\|_{\boldsymbol{\Sigma}_k} + \frac{\partial}{\partial \boldsymbol{z}}\big\|\boldsymbol{G}_{i,k}\boldsymbol{K}_k\big\|_{\boldsymbol{\Sigma}_k}(\boldsymbol{z}_k - \boldsymbol{z}_k^j)\Big), \quad \forall i \in \{1,2,..,l_u\}, \tag{4.11f}$$

$$|\boldsymbol{\kappa}_k - \boldsymbol{\kappa}_k^j| - \Omega^j \leq t_k, \quad -t_k \leq 0, \tag{4.11g}$$

$$\boldsymbol{s}_0 = \boldsymbol{s}(0), \tag{4.11h}$$

$$\boldsymbol{s}_f = \boldsymbol{s}(N), \tag{4.11i}$$

$$\forall k \in \{0,1,..,N-1\}. \tag{4.11j}$$

$\boldsymbol{z}_k \in \mathbb{R}^{9+3n_c} = (\boldsymbol{s}_k, \boldsymbol{v}_k)$ is the concatenated vector of states and controls at time k. Constraints (4.11e)-(4.11f) are the linearized state and control chance constraints, where

$$\frac{\partial}{\partial \boldsymbol{z}} \left\| \boldsymbol{H}_{i,k} \right\|_{\boldsymbol{\Sigma}_k} = \frac{1}{2 \left\| \boldsymbol{H}_{i,k} \right\|_{\boldsymbol{\Sigma}_k}} \Big(2 \boldsymbol{H}_i^\top \boldsymbol{\Sigma}_k \frac{\partial}{\partial \boldsymbol{z}} \boldsymbol{H}_{i,k} + \sum_{i=0}^{n} \sum_{j=0}^{n} h_i h_j \frac{\partial}{\partial \boldsymbol{z}} \Sigma_{ij} \Big). \tag{4.12}$$

$\frac{\partial}{\partial \boldsymbol{z}} \boldsymbol{\Sigma} \in \mathbb{R}^{9 \times 9 \times (9+3n_c)}$ represents the covariance derivative w.r.t. $\boldsymbol{z}$. Notice that this term is more involved since it includes the propagation of the tensor derivatives of the covariance matrix given the current states and controls as well as the previous states and controls as follows:

$$\frac{\partial}{\partial \boldsymbol{z}} \boldsymbol{\Sigma}_{k+1} = \sum_{i=0}^{k-1} \boldsymbol{A}_k \frac{\partial}{\partial \boldsymbol{z}} \boldsymbol{\Sigma}_{k+1|i} \boldsymbol{A}_k^\top + \frac{\partial}{\partial \boldsymbol{z}} \boldsymbol{\Sigma}_{k+1|k}. \tag{4.13}$$

We resort to the autodiff library JAX [9] for such computation. Finally, the trust region constraints (4.11g) are enforced only on the angular momentum $\boldsymbol{\kappa}_k$ since it's the only nonlinear part in the centroidal dynamics.

4.2 Simulations Results

In this section, we report simulation results for the quadruped robot Solo in the Pybullet simulation environment [21]. We compare trajectories generated using centroidal stochastic trajectory optimization against nominal trajectory optimization for trotting and bounding gaits on challenging unknown cluttered terrains. Offline, we warm-start the centroidal SCP solver using centroidal trajectories coming from the whole-body DDP solver Croccodyl [65]. Then, we optimize whole-body trajectories to track back the optimized centroidal and force trajectories from the SCP solver as illustrated pictorially in Fig. 4.1. The cost weights for both whole-body DDP and centroidal SCP are summarized in Table 4.1 and Table 4.2, respectively. Both DDP and SCP solvers were discretized with a sampling time of $\Delta_k = 10$ ms for a planning horizon length of $N = 165$, and motion plans were designed on a flat ground with a floor static coefficient of friction $\mu = 0.5$ for both solvers. During simulation (i.e. online), whole-body DDP joint-space trajectories were tracked at a higher sampling rate of $\Delta_k = 1$ ms using a PD control law:

$$\boldsymbol{\tau}_k = \bar{\boldsymbol{\tau}}_k + \boldsymbol{K}_p(\boldsymbol{q}_k - \tilde{\boldsymbol{q}}_k) + \boldsymbol{K}_d(\dot{\boldsymbol{q}}_k - \dot{\tilde{\boldsymbol{q}}}_k), \tag{4.14}$$

where $\bar{\boldsymbol{\tau}}_k$ are the DDP optimal feedforward joint torque controls, $\boldsymbol{q}_k$ and $\dot{\boldsymbol{q}}_k$ represent the DDP optimal joint positions and velocities respectively. Although in theory the optimal DDP gains could be used, it was not transferable in our case for highly

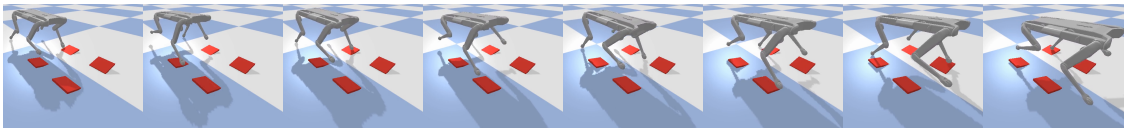


Figure 4.2: Trot motion in an unplanned cluttered environment using stochastic centroidal SCP and whole-body DDP.

Table 4.1: Whole-body DDP cost weights.

	DDP solver task weights	
Task	Trot	Bound
Swing foot	1e6	1e6
SCP CoM tracking	1e3	1e1
SCP centroidal tracking	1e3	1e3
SCP force tracking	1e2	8e1
Friction cone	2e2	2e0
State regulation	1e-1	1e-1
Control regulation	1e0	1e0
Contact impact velocity regulation	2e1	2e1

dynamic motions, especially with long horizons as the motion diverged quickly. For that reason, we used hand-tuned PD gains for the scenarios described in the following subsection. The chance-constraints hyper-parameters of the stochastic SCP were tuned as follows for the trotting and bounding motions: the probability level of friction pyramid constraint violations for every leg is $\alpha_u = 0.1$. The covariance of the contact position parametric uncertainties for each foot is set to $\boldsymbol{\Sigma}_\theta = \text{diag}\left[0.4^2, 0.4^2, 0.4^2\right]$. The covariance of additive centroidal uncertainties is set to $\boldsymbol{\Sigma}_w = \text{diag}\left[0.85^2, 0.4^2, 0.01^2, 0.75^2, 0.4^2, 0.01^2, 0.85^2, 0.4^2, 0.01^2\right]$, and $\boldsymbol{\Sigma}_w = \text{diag}\left[0.75^2, 0.4^2, 0.01^2, 0.85^2, 0.4^2, 0.01^2, 0.75^2, 0.4^2, 0.01^2\right]$ for the trotting and bounding motions, respectively.

4.2.1 Simulations setup

We ran a set of Monte-Carlo simulations for two scenarios per motion: Scenario 1) **without debris**: 100 simulations on flat ground with a reduced floor friction $\mu = 0.4$, while applying random lateral force disturbances for 200 ms at the center of the robot's base link. For trotting motion, we set $\boldsymbol{K}_p = 4.0 * \mathbb{I}$, $\boldsymbol{K}_d = 0.2 * \mathbb{I}$. For the bounding motion, we set $\boldsymbol{K}_p = 3.0 * \mathbb{I}$, $\boldsymbol{K}_d = 0.2 * \mathbb{I}$. Scenario 2) **with debris**:

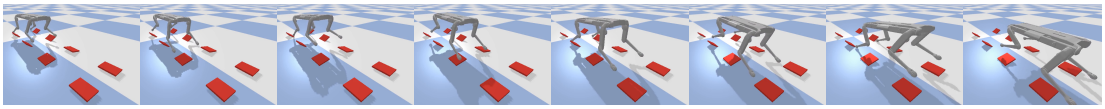


Figure 4.3: Bound motion in an unplanned cluttered environment using stochastic centroidal SCP and whole-body DDP.

Table 4.2: Centroidal SCP cost weights.

	SCP solver task weights	
Task	Trot	Bound
DDP CoM tracking	1e4	1e4
DDP linear momentum tracking	1e3	1e3
DDP angular momentum tracking	1e5	1e5
Lateral force regulation per foot (x-direction)	1e2	1e2
Lateral force regulation per foot (y-direction)	1e0	1e2
Vertical force regulation per foot	1e1	1e1
Initial trust region weight	1e2	1e2

100 simulations with reduced floor friction $\mu = 0.4$, while adding unplanned debris of $2-3$ cm height ($6.6-10\%$ of the robot's leg length) with varying orientations of $0-17$ degrees along x and y directions as shown in Fig. 4.2 and Fig. 4.3 for trotting and bounding motions, respectively (please refer to the video for more details).The joint impedances were set to $\boldsymbol{K}_p = 5.0 * \mathbb{I}$, $\boldsymbol{K}_d = 0.2 * \mathbb{I}$ for the trotting motion, and $\boldsymbol{K}_p = 4.7 * \mathbb{I}$, $\boldsymbol{K}_d = 0.2 * \mathbb{I}$ for the bounding motion. Further, we apply again random lateral force impulses for 200 ms at the center of the robot's base. The same force disturbances were applied to the nominal and stochastic trajectories and were sampled from a Gaussian distribution with zero mean and $\sigma = 15$ N (60% of the robot weight). The force impulse is applied at the same randomly sampled time instance after the first second of the motion. We analyze the robustness of the motions generated using stochastic SCP against their nominal counterpart by evaluating the Normalized cumulative sum of the contact position deviations of the robot feet when a foot is in contact with the ground (i.e. foot slippage), which reflects the saturation of the friction pyramid constraints. The normalized cumulative sum was computed by subtracting the average cumulative sum of the previous samples from the current integral quantity at each point in time. Moreover, we report the centroidal tracking performance between the generated SCP references and the simulated trajectories.

First, we discuss the optimized contact forces generated using nominal and stochastic SCPs, which are later tracked using whole-body DDP. In Fig. 4.4, we

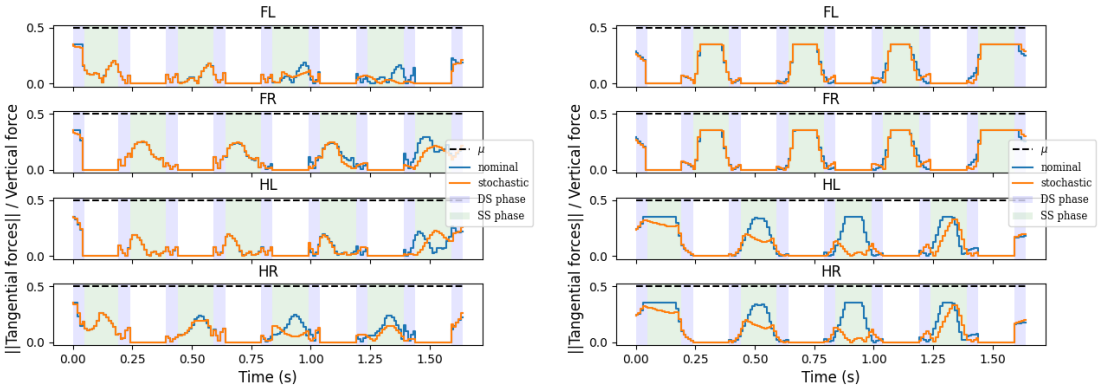


Figure 4.4: Ratio of norm of tangential forces w.r.t. vertical force for a trotting motion (left) and a bounding motion (right).

plot the ratio between the norm of the tangential forces and the vertical forces for dynamic trotting and bounding motions. As expected, the forces optimized using stochastic SCP saturate less the friction cones compared to the ones optimized with nominal SCP, especially during single support phases where the solution of the QP is unique. This highlights the contribution of the control back-off magnitudes, which increase along the horizon due to the covariance propagation along the linearized dynamics (4.11f).

For the trotting motion (Fig. 4.5), trajectories designed using stochastic SCP (our method) achieved less feet slippage mean (26.3% and 28.9% for scenario 1 and 2, respectively) than nominal SCP and an improved centroidal tracking performance mean (8.41% and 13.0%). The same analysis was carried out for the bounding motion in a more challenging terrain (Fig. 4.3). As shown in Fig. 4.6, stochastic SCP trajectories contributed to fewer feet slippage mean (22.8% and 14.8% for scenarios 1 and 2, respectively) than nominal SCP, and an improved centroidal tracking performance mean (25.6% and 13.6%).

4.3 Conclusions

In this work, we used nonlinear stochastic trajectory optimization for generating robust centroidal momentum trajectories for legged robots that take into account additive uncertainties on the centroidal dynamics as well as parametric uncertainties on the contact positions. We used a linearization-based covariance propagation for resolving the stochastic nonlinear dynamics. Furthermore, we resolved the friction pyramid joint chance constraints by designing proper upper bounds (back-offs) at each point in time on the individual hyper-planes forming the friction pyramid

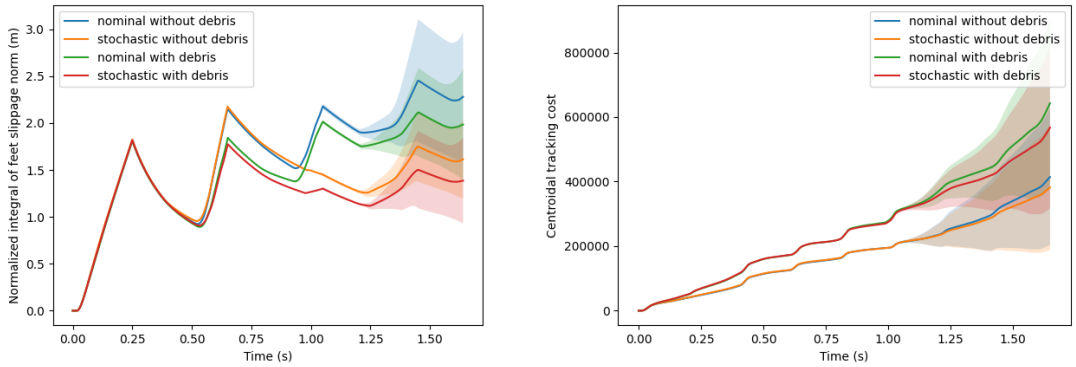


Figure 4.5: Normalized cumulative sum of feet slippage norm (left) and centroidal tracking cost (right) for a trotting motion.

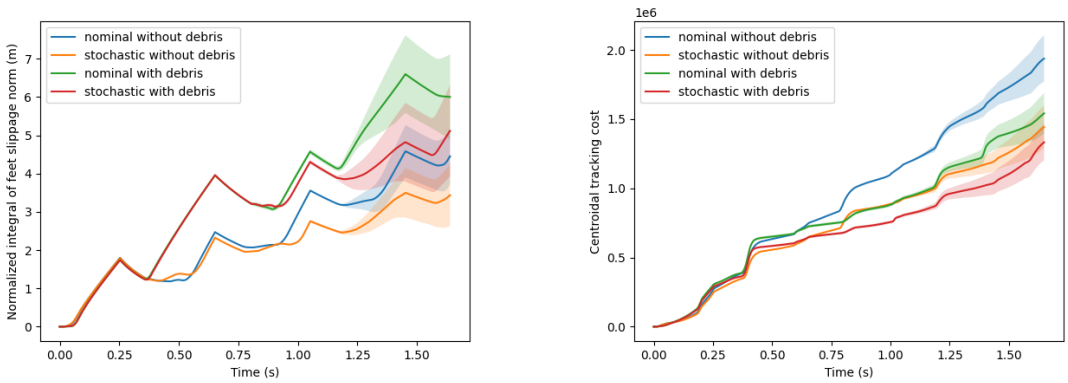


Figure 4.6: Normalized cumulative sum of feet slippage norm (left) and centroidal tracking cost (right) for a bounding motion.

polytopes. Finally, we presented a whole-body trajectory optimization framework that alternates between stochastic centroidal trajectory optimization and whole-body trajectory optimization for generating feasible robust whole-body motions. We used our framework to generate trotting and bounding dynamic gaits for the quadruped robot Solo. We then tracked these trajectories in a Pybullet physics simulator, introducing different disturbance realizations and contact uncertainties. The results show that our approach generated safer motions by contributing to less average contact slippage, as well as improved centroidal tracking performance over deterministic trajectory optimization. Although the current stochastic SCP approach does not require additional optimization variables over a deterministic approach SCP, the computational complexity is relatively higher due to uncertainty

propagation and the additional tensor derivatives required for solving the linearized chance constraints.

Another limitation of the current stochastic SCP approach lies in the accuracy of uncertainties propagation through the linearized dynamics, which might be hindered for long horizons. However, we believe that this might not be an issue in practice when applied in a receding horizon fashion. To this end, we plan to extend the current framework to nonlinear stochastic MPC in the future.

Chapter 5

Whole-Body Nonlinear Stochastic MPC

This chapter builds upon this publication [32]. Trajectory optimization under uncertainties is a challenging problem for robots in contact with the environment. Such uncertainties are inevitable due to estimation errors, control imperfections, and model mismatches between planning models used for control and the real robot dynamics. This induces control policies that could violate the contact location constraints by making contact at unintended locations and as a consequence leading to unsafe motion plans. This work addresses the problem of robust kino-dynamic whole-body trajectory optimization using stochastic nonlinear model predictive control (SNMPC) by considering additive uncertainties on the model dynamics subject to contact location chance constraints as a function of the robot's full kinematics. We demonstrate the benefit of using SNMPC over classic nonlinear MPC (NMPC) for whole-body trajectory optimization in terms of contact location constraint satisfaction (safety). We run extensive Monte-Carlo simulations for a quadruped robot performing agile trotting and bounding motions over small stepping stones, where contact location satisfaction becomes critical. Our results show that SNMPC can perform all motions safely with 100% success rate, while NMPC failed 48.3% of all motions.

5.1 Stochastic Optimal Control for Kinodynamic Trajectory Optimization

We present the stochastic OCP version of problem (2). Consider the following controlled stochastic diffusion:

$$\partial \boldsymbol{x} = \partial \boldsymbol{f}(\boldsymbol{x}_t, \boldsymbol{u}_t)\partial t + \boldsymbol{C}\partial \boldsymbol{w}_t, \tag{5.1}$$

where the state $\boldsymbol{x}_t \in \mathbb{R}^n$ evolves stochastically based on the additive random variable $\boldsymbol{w}_t \in \mathbb{R}^n$. The selector matrix $\boldsymbol{C} \in \mathbb{R}^{n \times n}$ maps the additive disturbance on the dynamics.

Assumption 4. *(Additive disturbance process)*
$\partial \boldsymbol{w}_t \sim \mathcal{N}(\boldsymbol{0}, \partial t)$ *is an additive Gaussian random process with zero mean.*

Assumption 5. *(State feedback control policy)*
$\boldsymbol{u}_t \in \mathbb{R}^m$ *is a causal state feedback control policy in the form of* $\boldsymbol{u}_t = \boldsymbol{u}_t^* + \boldsymbol{K}(\boldsymbol{x}_t^* - \boldsymbol{x}_t)$, *where* $\boldsymbol{u}_t^*$ *is the optimized feedforward open-loop control actions,* $\boldsymbol{K} \in \mathbb{R}^{m \times n}$ *are stabilizing feedback gains, and* $\boldsymbol{x}_t$ *is the deterministic state evolving as* $\boldsymbol{x}_{t+1} = \boldsymbol{f}(\boldsymbol{x}_t, \boldsymbol{u}_t)$.

Given the above, we aim to solve the following SNMPC problem with contact location chance constraints.

Problem 6. *Kino-dynamic SNMPC problem with contact location joint chance-constraints*

$$\underset{\boldsymbol{X},\boldsymbol{U},\boldsymbol{S}}{\text{minimize}} \;\; \mathbb{E}\left[\mathcal{L}_{\text{total}}(\boldsymbol{X}, \boldsymbol{U}, \boldsymbol{S})\right] \tag{5.2a}$$

$$\text{subject to} \tag{5.2b}$$

$$\boldsymbol{f}'_{\text{impl}}\left(\boldsymbol{x}_{k+1}, \boldsymbol{x}_k, \boldsymbol{u}_k, \boldsymbol{w}_k\right) = \boldsymbol{0}, \tag{5.2c}$$

$$(2.14), (2.15), (2.16a), \tag{5.2d}$$

$$\Pr(2.16b) \geq \alpha_{i,k}, \quad \forall i \in \{1, \ldots, n_c\}, \tag{5.2e}$$

$$-\boldsymbol{s}_k \leq \boldsymbol{0}, \qquad \forall k \in \{0, 1, \ldots, N-1\}, \tag{5.2f}$$

$$\boldsymbol{x}_0 - \boldsymbol{x}(t) = \boldsymbol{0}, \tag{5.2g}$$

where (5.2c) are the discrete-time implicit stochastic dynamics equality path constraints in (5.1). The constraints (5.2d) are to be satisfied deterministically by enforcing them on the mean of the state. In this work, we aim to account for the additive uncertainties by enforcing the contact location constraints in the $x-y$ directions probabilistically within at least a probability level $\alpha_{i,k}$ (5.2e), which are known as chance-constraints. The above SNMPC problem is not tractable in general because the dynamics are stochastic, and resolving the chance-constraints (5.2e) requires the integration of multi-dimensional Probability Density Functions (PDFs), which becomes computationally intractable for high dimensions. To tackle those issues, we solve an approximate deterministic reformulation of the above OCP.

5.1.1 Tractable formulation of Friction Pyramid Joint Chance-constraints

The goal of the following subsections is to design safety margins/upper bounds known as *back-offs* on the contact location chance-constraints (5.2e) to accommodate for the additive stochastic disturbances on the dynamics that are difficult for only feedback to deal with. This is particularly crucial for legged robots since making contact at unintended contact locations can lead to unsafe motions. We design such back-offs formally based on the evolution of statistical information along the horizon inside the optimization problem, such that we can provide probabilistic statements about constraint satisfaction without degrading the performance. Notice that those margins are not fixed compared to designing them heuristically by hand (check the results section). In other words, if the variance of the uncertainty is large, or we want to satisfy the constraints with a larger probability, then it reflects automatically on the back-off magnitude by increasing the safety margin accordingly to ensure the expected probability of constraint satisfaction. To reduce the computational complexity for solving the contact-location joint chance-constraints (5.2e), we first linearize the nonlinear constraints around the j-th SQP iteration at $\Delta \boldsymbol{x}_k \triangleq \boldsymbol{x}_k - \boldsymbol{x}_k^j$, then solve for each half-space chance-constraints forming the linearized feasible set as follows:

$$\nabla_{\boldsymbol{x}_k}(2.16\text{b}) = \gamma_{i,k}.\Big(\boldsymbol{h}^{x,y}_{\text{pos}_{i,k}}(\tilde{\boldsymbol{q}}_k^j) + \boldsymbol{J}^{x,y}_{i,k}(\tilde{\boldsymbol{q}}_k^j)\Delta\boldsymbol{x}_k \in \mathcal{S}^{x,y}_{i,k}\Big). \tag{5.3}$$

Given the above-linearized constraints, we can write down the contact location chance constraints as a conjunction of half-space constraints of the 2D polygon forming $\mathcal{S}^{x,y}_{i,k} \in \mathbb{R}^4$ as

$$\begin{aligned} &\Pr\left(\bigwedge_{l=1}^{4} \boldsymbol{G}^l_{i,k}(\tilde{\boldsymbol{q}}_k^j)\boldsymbol{x}_k + g^l_{i,k} \in \mathcal{S}^{x,y}_{i,k}\right) \geq \alpha_{i,k} \\ &\equiv \Pr\left(\bigvee_{l=1}^{4} \boldsymbol{G}^l_{i,k}(\tilde{\boldsymbol{q}}_k^j)\boldsymbol{x}_k + g^l_{i,k} \notin \mathcal{S}^{x,y}\right) \leq 1 - \alpha_{i,k}, \end{aligned} \tag{5.4}$$

where $\boldsymbol{G}^l_{i,k}$ is the l-th row of $\pm\boldsymbol{J}^{x,y}_{i,k}(\tilde{\boldsymbol{q}}_k^j) \in \mathbb{R}^{4\times n}$, and $g^l_{i,k}$ is the l-th element of the vector $\boldsymbol{g}_{i,k} \pm \left(\boldsymbol{J}^{x,y}_{i,k}\boldsymbol{x}_k^j - \boldsymbol{h}^{x,y}_{\text{pos}_{i,k}}\right)$. To avoid multi-dimensional integrals of joint chance constraints, we use Boole's inequality

$$\Pr(\bigvee_{i=1}^{n} \boldsymbol{C}_i) \leq \sum_{i}^{n} \Pr(\boldsymbol{C}_i), \tag{5.5}$$

as a conservative union bound on the joint chance constraints [80]. By applying Boole's inequality we get

$$(5.4) \xleftarrow{(5.5)} \sum_{l=1}^{4} \Pr\left(\boldsymbol{G}_{i,k}^{l}(\tilde{\boldsymbol{q}}_k^j)\boldsymbol{x}_k > g_{i,k}^{l}\right) \leq 1 - \alpha_{i,k}. \tag{5.6}$$

By allocating equal risk for constraint violation for each half-space constraint $\epsilon_{i,k} \triangleq (1-\alpha_{i,k})/4$, we finally arrive to:

$$\begin{aligned}(5.6) &\Leftarrow \Pr\left(\boldsymbol{G}_{i,k}^{l}(\tilde{\boldsymbol{q}}_k^j)\boldsymbol{x}_k + g_{i,k}^{l} \notin \mathcal{S}^{x,y}\right) \leq \epsilon_{i,k}, \\ &\equiv \Pr\left(\boldsymbol{G}_{i,k}^{l}(\tilde{\boldsymbol{q}}_k^j)\boldsymbol{x}_k + g_{i,k}^{l} \in \mathcal{S}^{x,y}\right) \geq 1 - \epsilon_{i,k}, \\ &\forall i = 1,\dots,n_c,\ \forall k = 0,\dots,N-1,\ \forall l = 1,\dots,4. \end{aligned} \tag{5.7}$$

Given the above-linearized chance constraints, we proceed to solve for each unimodal PDF representing the half-space chance constraints forming the contact surface polygon.

5.1.2 Deterministic Reformulation of Individual Contact location Chance-Constraints

In this subsection, we present a tractable deterministic formulation for solving the above individual chance constraints (5.7). This requires statistical knowledge of the uncertainty propagation through nonlinear dynamics. One way to do this is by exploiting sampling methods like unscented-based transforms [85], or Generalized Polynomial Chaos (gPC) [77]. These methods can lead to a more accurate estimate of the propagated uncertainty at the cost of a significant increase in computational complexity, which is often too time-demanding for real-time MPC. Since computational efficiency is critical in our case, we defer from using these methods. Instead, we adopt an approximate linearization-based covariance propagation as in [121][57]. Based on assumptions (4)-(5), the approximate mean and covariance of the dynamics evolve as

$$\bar{\boldsymbol{x}}_{k+1} \approx \boldsymbol{f}(\bar{\boldsymbol{x}}_k, \bar{\boldsymbol{u}}_k) + \boldsymbol{A}_k \Delta\bar{\boldsymbol{x}}_k + \boldsymbol{B}_k \Delta\bar{\boldsymbol{u}}_k, \tag{5.8a}$$

$$\boldsymbol{\Sigma}_{\boldsymbol{x}_{k+1}} = \boldsymbol{A}_{\text{cl}} \boldsymbol{\Sigma}_{\boldsymbol{x}_k} \boldsymbol{A}_{\text{cl}}^{\top} + \boldsymbol{C}\boldsymbol{\Sigma}_{\boldsymbol{w}}, \quad \boldsymbol{\Sigma}_{\boldsymbol{x}_0} = \boldsymbol{0}, \tag{5.8b}$$

$$\boldsymbol{A}_{\text{cl}} \triangleq \boldsymbol{A}_k + \boldsymbol{B}_k \boldsymbol{K}_k. \tag{5.8c}$$

The initial condition is assumed to be deterministic (i.e. $\boldsymbol{x}_0 = \bar{\boldsymbol{x}}_0$), where the $\bar{}$ superscript denotes the mean of the quantity. The j-th controls perturbation is defined as $\Delta\bar{\boldsymbol{u}}_k \triangleq \bar{\boldsymbol{u}}_k - \bar{\boldsymbol{u}}_k^j$, and (5.8c) is the linearized closed-loop dynamics, where

$\boldsymbol{A}_k \triangleq \nabla_{\bar{\boldsymbol{x}}} \boldsymbol{f}(\bar{\boldsymbol{x}}_k, \bar{\boldsymbol{u}}_k)|_{\bar{\boldsymbol{x}}_k^j, \bar{\boldsymbol{u}}_k^j}$, and $\boldsymbol{B}_k \triangleq \nabla_{\bar{\boldsymbol{u}}} \boldsymbol{f}(\bar{\boldsymbol{x}}_k, \bar{\boldsymbol{u}}_k)|_{\bar{\boldsymbol{x}}_k^j, \bar{\boldsymbol{u}}_k^j}$ are the Jacobians of the dynamics w.r.t. the nominal state and control respectively. The feedback gains $\boldsymbol{K}_k$ are computed using the Discrete Algebraic Riccati Equation (DARE) based on the time-varying linearized dynamics. Using the covariance information in (5.8b) and the probability level of constraint violation $\epsilon_{i,k}$, we seek the least conservative upper bound $\eta_{i,k}^l$ at each point in time for each half-space contact location chance-constraint (5.7). By exploiting the inverse of the *Cumulative Density Function* (CDF) Φ^{-1} of a Gaussian distribution, we get a deterministic reformulation of the individual contact location chance-constraints:

$$\boldsymbol{G}_{i,k}^l(\tilde{\boldsymbol{q}}_k^j)\bar{\boldsymbol{x}}_k + g_k^l \in \mathcal{S}^{x,y} - \eta_k^l, \quad \forall l = 1, \dots, 4, \tag{5.9a}$$

$$\eta_k^l \triangleq \Phi^{-1}(1 - \epsilon_k^l) \left\| \boldsymbol{G}_i^l \right\|_{\boldsymbol{\Sigma}_{\boldsymbol{x}_k}}. \tag{5.9b}$$

η_k^l is the back-off bound that ensures the satisfaction of the individual chance constraints (5.7) illustrated pictorially in Fig. 5.1. Contrary to designing a heuristic-based back-off bound by hand, this upper bound is not fixed (i.e. it varies at every point in time along the horizon). This is because the magnitude of this back-off bound is scaled by the covariance propagation along the horizon (5.8b), the magnitude of the time-varying feedback gain $\mathbf{K}_k$ (5.8c), as well as the design parameter $\epsilon_{i,k}$ capturing the desired probability of constraint satisfaction. Although in theory, one can optimize for both $\boldsymbol{K}_k$ and $\epsilon_{i,k}$ for better performance, this usually leads to a bi-level optimization, which is computationally expensive to solve in real-time for MPC applications.

5.1.3 Deterministic Reformulation of SNMPC

Given the previous reformulation of the individual chance constraints, we can write down a deterministic reformulation of the original SNMPC problem (6) on the mean of the nonlinear dynamics. Despite the reformulated chance-constraints constraints, this problem has the same number of optimization variables as the NMPC problem (2), which means that with the gained robustness, we don't increase the computational complexity of the problem over NMPC.

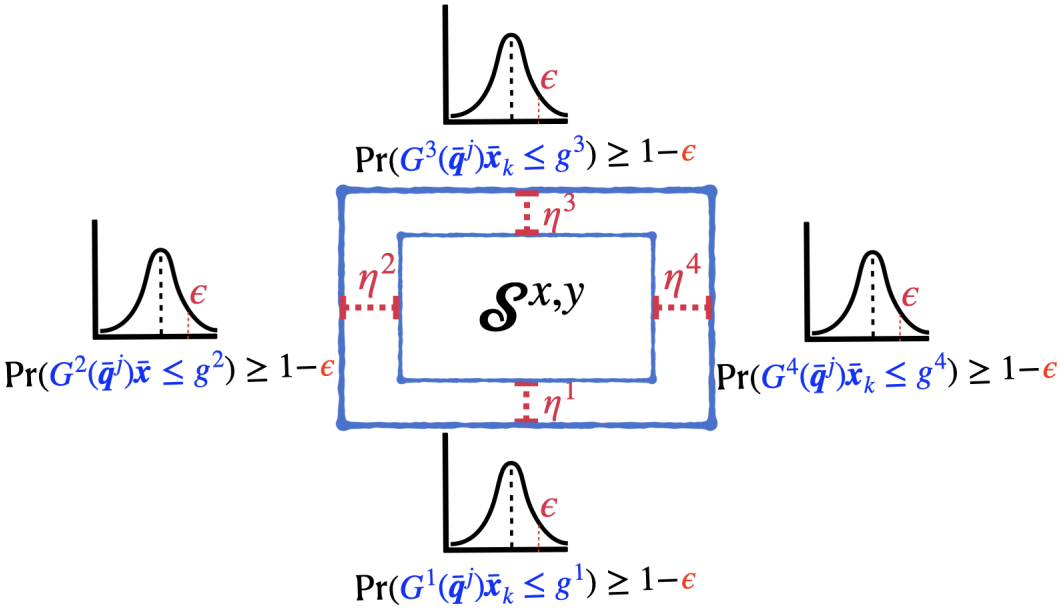


Figure 5.1: Effect of equally distributed back-offs design of the linearized contact location chance-constraints.

Problem 7. *Kino-dynamic SNMPC problem with individual chance constraints*

$$\underset{\bar{\boldsymbol{X}},\bar{\boldsymbol{U}},\boldsymbol{S}}{\text{minimize}} \;\; \mathcal{L}_{\text{total}}(\bar{\boldsymbol{X}},\bar{\boldsymbol{U}},\boldsymbol{S}) \tag{5.10a}$$

$$\text{s.t.} \tag{5.10b}$$

$$\boldsymbol{F}(\bar{\boldsymbol{x}}_{k+1},\bar{\boldsymbol{x}}_k,\bar{\boldsymbol{u}}_k) \triangleq \boldsymbol{f}_{\text{impl}}(\bar{\boldsymbol{x}}_k,\bar{\boldsymbol{x}}_{k+1},\bar{\boldsymbol{u}}_k) = \boldsymbol{0}, \tag{5.10c}$$

$$\boldsymbol{E}(\bar{\boldsymbol{x}}_k,\bar{\boldsymbol{u}}_k) \triangleq \begin{cases} \boldsymbol{h}_{\text{eq}}(\bar{\boldsymbol{x}}_k,\bar{\boldsymbol{u}}_k,\boldsymbol{s}_k) = \boldsymbol{0}, \\ \bar{\boldsymbol{x}}_0 - \boldsymbol{x}(t) = \boldsymbol{0}, \end{cases} \tag{5.10d}$$

$$\boldsymbol{I}(\bar{\boldsymbol{x}}_k,\bar{\boldsymbol{u}}_k) \triangleq \begin{cases} \boldsymbol{h}_{\text{ineq}}(\bar{\boldsymbol{x}}_k,\bar{\boldsymbol{u}}_k,\boldsymbol{s}_k) \leq \boldsymbol{0}, \\ \boldsymbol{G}(\bar{\boldsymbol{x}}_k^j)\bar{\boldsymbol{x}}_k + \boldsymbol{g}_k + \boldsymbol{J}_{\text{sg}}\boldsymbol{s}_k \in \mathcal{S}^{x,y} - \boldsymbol{\eta}_k, \end{cases} \tag{5.10e}$$

$$-\boldsymbol{s}_k \leq \boldsymbol{0}, \quad \forall k \in \{0,1,\ldots,N-1\}. \tag{5.10f}$$

This SMPC problem optimizes for the open-loop mean states $\bar{\boldsymbol{X}} = \{\bar{\boldsymbol{x}}_0,\ldots,\bar{\boldsymbol{x}}_N\}$, and feedforward controls $\bar{\boldsymbol{U}} = \{\bar{\boldsymbol{u}}_0,\ldots,\bar{\boldsymbol{u}}_N\}$. All nonlinear equality constraints are captured inside $\boldsymbol{h}_{\text{eq}}(.) = \left[(2.14)^\top,(2.16a)^\top,(2.17)^\top\right]^\top$, while $\boldsymbol{h}_{\text{ineq}}(.) =$ (2.15) captures the friction cone inequality constraints. Finally, the second row of the inequality constraints (5.10e) are the backed-off contact location constraints in the $x-y$ directions, where $\boldsymbol{\eta_k} = \eta_k.\boldsymbol{1}_{2n_c}$. These linearized constraints are implemented softly with $\boldsymbol{J}_{\text{sg}}$ being the slack selector matrix. The above OCP is solved using

Algorithm 1: Approximate SNMPC Algorithm.

$\boldsymbol{Z}^0_{\text{traj}} \leftarrow$ Offline whole-body iLQR initial guess

for $j = 0, 1, \ldots, N_{\text{traj}} - 1$ (trajectory horizon) **do**

`/* QP PREPARATION PHASE */`

$\boldsymbol{Z}^j \leftarrow$ **initial guess** $\begin{cases} \boldsymbol{Z}^0_{\text{traj}} & \text{if } j = 0 \\ \boldsymbol{Z}^{j-1}_{\text{mpc}} & \text{otherwise} \end{cases}$

for $k = 0, 1, \ldots, N_{\text{mpc}} - 1$ (MPC horizon) **do**

$\boldsymbol{A}^j_k, \boldsymbol{B}^j_k \leftarrow$ computeFWDSensitivities(5.8a)

$\boldsymbol{K}^j_k \leftarrow$ computeDARE($\boldsymbol{A}^j_k, \boldsymbol{B}^j_k, \boldsymbol{Q}, \boldsymbol{R}$)

$\boldsymbol{\Sigma}^j_k \leftarrow$ propagateCovariance(5.8b) – (5.8c)

$\boldsymbol{\eta}^j_k \leftarrow$ computeBackOffs(5.9a)

`/* FEEDBACK PHASE */`

$\boldsymbol{\Delta Z}^*, \boldsymbol{S}^* \leftarrow$ solveQPSubProblem 8

$\boldsymbol{u}^j_0 \leftarrow$ FeebackPolicyAssumption 5

$\mathbf{Z}^{j+1} \leftarrow \mathbf{Z}^j + \Delta \mathbf{Z}^*, \ \boldsymbol{\zeta}^{j+1} = \boldsymbol{\zeta}^*$

$\boldsymbol{\beta}^{j+1} \leftarrow \boldsymbol{\beta}^*, \ \boldsymbol{\gamma}^{j+1} \leftarrow \boldsymbol{\gamma}^*$

Output: $\mathbf{X}^*_{\text{traj}}, \mathbf{U}^*_{\text{traj}}, \mathbf{S}^*_{\text{traj}}$

Sequential Quadratic Programming (SQP) [79] by constructing a quadratic model of the cost objective subject to linearized constraints that solves the Karush-Kuhn-Tucker (KKT) system associated with the following Lagrangian:

$$\Psi(\boldsymbol{z}) = \mathcal{L}_{\text{total}} + \boldsymbol{\zeta}^\top \boldsymbol{F} + \boldsymbol{\beta}^\top \boldsymbol{E} + \boldsymbol{\gamma}^\top \boldsymbol{I}, \tag{5.11}$$

where $\boldsymbol{z} \triangleq [\bar{\boldsymbol{x}}^\top, \bar{\boldsymbol{u}}^\top]^\top$ is the concatenated vector of states and controls. $\boldsymbol{\zeta}$, $\boldsymbol{\beta}$ are the associated Lagrange multipliers to the equality constraints, and $\boldsymbol{\gamma}$ are the ones corresponding to the inequality constraints. Given a perturbation $\Delta \boldsymbol{z}_k \triangleq \boldsymbol{z}_k - \boldsymbol{z}^j_k$, where $\boldsymbol{z}^j$ is the current initial guess along the control horizon, the following QP subproblem is solved:

Problem 8. *QP subproblem*

$$\underset{\Delta \boldsymbol{Z}, \boldsymbol{S}}{\text{minimize}} \ \frac{1}{2}\Delta \boldsymbol{z}^{j^\top} \boldsymbol{H} \Delta \boldsymbol{z}^j + \boldsymbol{p}^\top \Delta \boldsymbol{z}^j \tag{5.12a}$$

$$\text{s.t.} \tag{5.12b}$$

$$\boldsymbol{F}(\boldsymbol{z}^j) + \nabla_{\boldsymbol{z}} \boldsymbol{F}(\boldsymbol{z}^j) \Delta \boldsymbol{z}^j = \boldsymbol{0}, \tag{5.12c}$$

$$\boldsymbol{E}(\boldsymbol{z}^j) + \nabla_{\boldsymbol{z}} \boldsymbol{E}(\boldsymbol{z}^j) \Delta \boldsymbol{z}^j = \boldsymbol{0}, \tag{5.12d}$$

$$\boldsymbol{I}(\boldsymbol{z}^j) \ + \nabla_{\boldsymbol{z}} \boldsymbol{I}(\boldsymbol{z}^j) \Delta \boldsymbol{z}^j \leq \boldsymbol{0}, \tag{5.12e}$$

$$\boldsymbol{s}^j \leq \boldsymbol{0}. \tag{5.12f}$$

The Hessian of the Lagrangian $\boldsymbol{H} \triangleq \nabla^2_{\boldsymbol{z}} \Psi(\boldsymbol{z}^j)$ is approximated using the Generalized Gauss-Newton (GGN) variant of SQP as $\boldsymbol{H} \approx \nabla^\top_{\boldsymbol{z}} \Psi(\boldsymbol{z}^j) \nabla_{\boldsymbol{z}} \Psi(\boldsymbol{z}^j)$, and the gradient of the residual is defined as $\boldsymbol{p} \triangleq \nabla^\top_{\boldsymbol{z}} \Psi(\boldsymbol{z}^j) \Psi(\boldsymbol{z}^j)$. For an exact SQP iteration, the linearization of the backed-off contact location constraints included in the above inequality constraints includes the extra derivative $\nabla_{\boldsymbol{z}} \eta(\boldsymbol{z}^j_k)$:

$$\underbrace{\boldsymbol{G}^l_{i,k}(\bar{\boldsymbol{x}}^j_k)\bar{\boldsymbol{x}}_k \leq g^l_{i,k} - \eta^l_k}_{\boldsymbol{I}_{\boldsymbol{G}}(\boldsymbol{z}^j_k)} - \underbrace{\Phi^{-1}(1-\epsilon^l_{i,k})\Big(\nabla_{\boldsymbol{z}} \Big\| \boldsymbol{G}^l_{i,k}(\bar{\boldsymbol{x}}^j_k) \Big\|_{\boldsymbol{\Sigma}_{\boldsymbol{x}_k}} (\boldsymbol{z}_k - \boldsymbol{z}^j_k)\Big)}_{\nabla_{\boldsymbol{z}} \eta(\boldsymbol{z}^j_k)},$$

$$\nabla_{\boldsymbol{z}} \Big\| \boldsymbol{G}^l_{i,k}(\bar{\boldsymbol{x}}^j_k) \Big\|_{\boldsymbol{\Sigma}_{\boldsymbol{x}_k}} \triangleq \big(2 \Big\| \boldsymbol{G}^l_{i,k} \Big\|_{\boldsymbol{\Sigma}_{\boldsymbol{x}_k}}\big)^{-1} \Big(2 \boldsymbol{G}^{l^\top}_{i,k} \boldsymbol{\Sigma}_{\boldsymbol{x}_k} \nabla_{\boldsymbol{z}} \boldsymbol{G}^l_{i,k} + \sum_{i=0}^{n} \sum_{j=0}^{n} g^l_i g^l_j \nabla_{\boldsymbol{z}} \Sigma_{ij}\Big),$$

$$\forall l \in \{1, \ldots, 4\}, \forall i \in \{1, \ldots, n_c\}, \forall k \in \{0, \ldots, N\}. \tag{5.13a}$$

The above derivative involves the tensor derivative of the covariance matrix $\nabla_{\boldsymbol{z}} \boldsymbol{\Sigma}_{\boldsymbol{x}_k}$, which is expensive to compute.

Remark 6. *For real-time computational tractability, we adopt a SQP-type iteration by approximating $\nabla_{\boldsymbol{z}} \eta(\boldsymbol{z}^j_k) = 0$ as in [44]. This SQP approach is sub-optimal because we don't compute the exact Jacobian of the contact location inequality constraint as in [33], [57]. Despite this sub-optimality, this scheme yields good results in practice without sacrificing computational complexity over NMPC.*

The OCP is implemented with real-time iteration [27], where one QP subproblem is solved at a time using a full Newton-type step without a line search (see Algorithm 1).

5.2 Simulations Results

We report simulation results comparing SNMPC against NMPC for the quadruped robot Solo [37] performing dynamic trotting and bounding gaits on non-coplanar

small stepping stones. The robustness of both controllers is tested in terms of contact location constraint satisfaction (safety), and performance computed using the least-squares tracking cost (2.18a). Moreover, we test the safety margins induced by SNMPC against heuristic-based NMPC (HNMPC) and NMPC. For HNMPC, we shrank the contact-location constraints heuristically by hand by performing a grid search on an interval between 1 cm and 3 cm. A safety margin of 3 cm was selected as it was the first value where the contact-location constraints became active and differed from NMPC.

We conducted two sets of simulations: A) **Kino-dynamic Monte-Carlo Simulations**, where we test the robustness of the kino-dynamic model against persistent disturbance realizations. B) **Whole-body simulations**, to test the effect of model mismatch between the kino-dynamic model and the whole-body model of the robot in the Pybullet simulator. All three MPC approaches follow a trajectory generated offline using whole-body DDP from the Croccodyl solver [65] with pre-planned contact locations at the center of the contact surfaces. Also, the first MPC iteration is warm-started using this trajectory, while subsequent MPC iterations are warm-started from the previous MPC solution. The cost weights for the kino-dynamic MPC are summarized in Table 5.1. All problems were discretized with a sampling time of $\Delta_k = 10$ ms for an MPC horizon length of $N = 40$, and $N = 55$ for the trot and bound motions respectively. The motion plans were designed with a coefficient of friction $\mu = 0.5$. Finally, the real-time iteration scheme was performed using the optimal control solver ACADOS [108], exploiting Casadi's automatic differentiation [1], and Pinocchio's analytical derivatives for rigid body kinematic functions for computing the underlying derivatives [15].

5.2.1 Kino-dynamic Monte-Carlo Simulations

We run 500 closed-loop kino-dynamic Monte-Carlo simulations for each motion (trotting and bounding). We sample additive kinematic disturbance realizations from a multi-variate Gaussian distribution with zero mean and a covariance $\boldsymbol{\Sigma}_w = \mathbf{DIAG}\,[\mathbf{0}_6, 0.3^2, 0.3^2, 0.3^2, 0.2^2, 0.2^2, 0.2^2, 0.7^2, 0.7^2, 0.7^2, 0.7^2, 0.7^2, 0.7^2, 0.7^2, 0.7^2, 0.7^2, 0.8^2, 0.8^2, 0.8^2, 0.1^2, 0.1^2, 0.1^2, 0.7^2, 0.7^2, 0.7^2, 0.7^2, 0.7^2, 0.7^2, 0.7^2, 0.7^2, 0.7^2]^\top$. We tune the risk of violating the contact location constraints for SNMPC to be $\epsilon = 0.01$ for all feet and contact surfaces. The disturbances are applied on the base velocity at the time contacts are made to mimic the effect of impacts on the kino-dynamic model, as well as on the swing leg joint velocities during take-off and landing to simulate persistent disturbances and control imperfections on the swing legs. Finally, no disturbances are applied at the feet after impact based on the assumption that the feet do not slip. The disturbance realizations are discretized and integrated on the dynamics (5.1) using the Implicit-Euler integration scheme.

We report the percentage of successful motions in Table 5.2a. As shown, SN-

Table 5.1: MPC cost weights.

Task	**Kin-dyn weights**	
	Trot	Bound
CoM tracking	2e3	2e3
linear momentum tracking	2e2	2e2
angular momentum tracking	2e4	2e4
base position tracking	2e1	2e1
base relative orientation regulation	2e2	2e2
joint positions tracking	1e3	1e3
base linear velocity tracking	2e2	2e2
base angular velocity tracking	6e2	2e2
joint velocities tracking	8e1	6e1
force regulation (x-direction)	1e1	1e2
force regulation (y-direction)	1e1	2e1
force regulation (z-direction)	2e0	2e0
joints acceleration regulation	6e-3	1e-2
Constraint	**Slack L1/L2 weights**	
	Trot	Bound
friction cones constraint	5e0/5e-1	1e3/0e0
foot velocity equality constraint	5e2/0e0	1e4/0e0
CoM kin-dyn equality constraint	0e0/5e1	0e0/5e1
lin. mom. kin-dyn equality constraint	0e0/1e1	0e0/1e1
ang. mom. kin-dyn equality constraint	0e0/1e1	0e0/1e1
contact location chance-constraints (x-y)	1e4/0e0	1e4/0e0
contact location chance-constraint (z)	5e4/0e0	3e4/0e0

MPC manages to perform all motions successfully without violating any of the contact location constraints despite the disturbances, which satisfies the expected probability of constraint satisfaction (99%) thanks to the design of contact location constraints back-off design in (4.7a). On the contrary, NMPC violated the contact location constraints 48.3% of all motions. Finally, HNMPC violated fewer constraints than NMPC, but still worse than SNMPC despite the robustness induced by shrinking the constraint set by hand. We highlight that although this heuristic works fairly for the trot case (success rate of 85.4%), using the same metric performed worse for a more agile bounding motion with a success rate of 67%, which dictates that the user needs to keep tuning the controller blindly every time the OCP parameters changes to attain the desired empirical results. To quantify

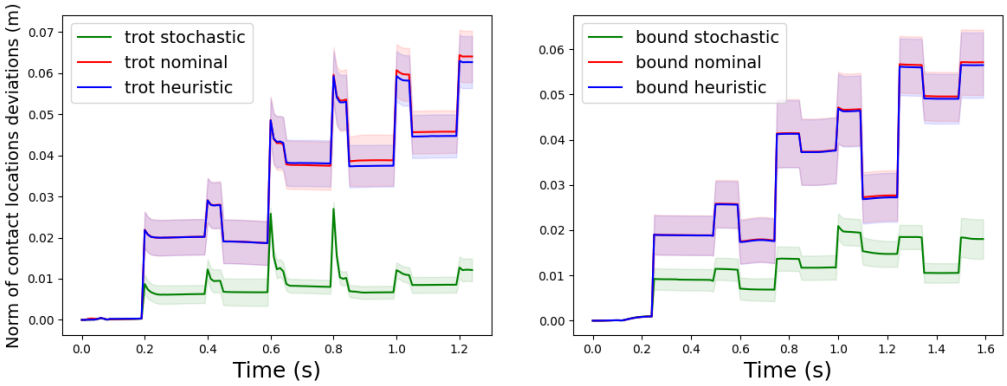
Table 5.2: Robustness and performance.

(a) Rate of successful motions

Task	NMPC	HNMPC	SNMPC
Trot	51.0%	85.4%	**100%**
Bound	52.4%	67%	**100%**

(b) Open-loop MPC cost.

Task	NMPC	HNMPC	SNMPC
Trot$\times 10^7$	**3.27**	3.28	3.31
Bound$\times 10^8$	**4.72**	**4.72**	4.77



(a) Trot motion. (b) Bound motion.

Figure 5.2: Norm of the contact location deviations from the contact surface center using NMPC, HNMPC, and SNMPC.

the safety margin induced by all controllers, we plot the mean and the 2σ distance between the end-effector positions and the center of the contact surface in Fig. 5.2 showing that SNMPC induced the best safety margin being the closest to the center of the contact surfaces.

Finally, we plot the performance of the three controllers in Table 5.2b based on open-loop MPC, where we plug the predicted open-loop state instead of the measured state during re-planning. We highlight that we didn't use Monte Carlo simulations in computing the performance due to presence of large number failed trajectories in both NMPC and HNMPC cases. Although SNMPC sacrifices a bit the performance for safety, the performance of the three controllers is comparable. This is because the swing foot tracking of the controllers is affected by real-time iteration schemes (non-full-convergence of the OCPs), and slack penalties on the constraints.

5.2.2 Whole-body Simulation

In this subsection, we test the effect of model mismatch between the kino-dynamic model and the whole-body model (i.e. the dynamic effects of the legs). Although in practice the legs are assumed to be massless for quadruped robots, their effect cannot be neglected for agile motions. Moreover, impulsive dynamics during impacts are also ignored since they are usually hard to model. Finally, since we are running real-time iterations, neither solver achieves full convergence in one Newton/Newton-type step. As a consequence, the previous effects can hinder the satisfaction of the contact location constraints. To test those effects, we report whole-body simulations of the quadruped robot Solo [37] in the Pybullet simulation environment [21] for dynamic trot and bound motions shown in Fig. 5.3 and Fig. 5.4 respectively. The whole-body simulation runs with a discretization time of $\Delta_{sim_k} = 1$ ms, where the feedforward MPC trajectories are linearly interpolated. We apply the following state feedback control law to both controllers:

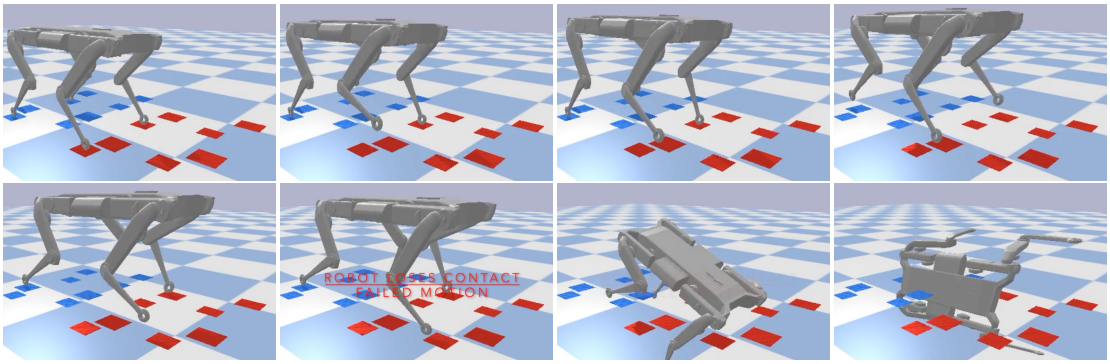
$$\boldsymbol{\tau}_k = \boldsymbol{\tau}_k^* + \boldsymbol{K}_p(\boldsymbol{q}_{j_k}^* - \boldsymbol{q}_{j_k}) + \boldsymbol{K}_d(\dot{\boldsymbol{q}}_{j_k} - \dot{\boldsymbol{q}}_{j_k}), \tag{5.14a}$$

$$\boldsymbol{\tau}_k^* \triangleq \text{RNEA}(\tilde{\boldsymbol{q}}_k^*, \dot{\boldsymbol{q}}_k^*, \ddot{\boldsymbol{q}}_k^*) - \sum_{i=0}^{n_c} \boldsymbol{J}_i^\top(\tilde{\boldsymbol{q}}_k^*)\boldsymbol{\lambda}_i^*. \tag{5.14b}$$

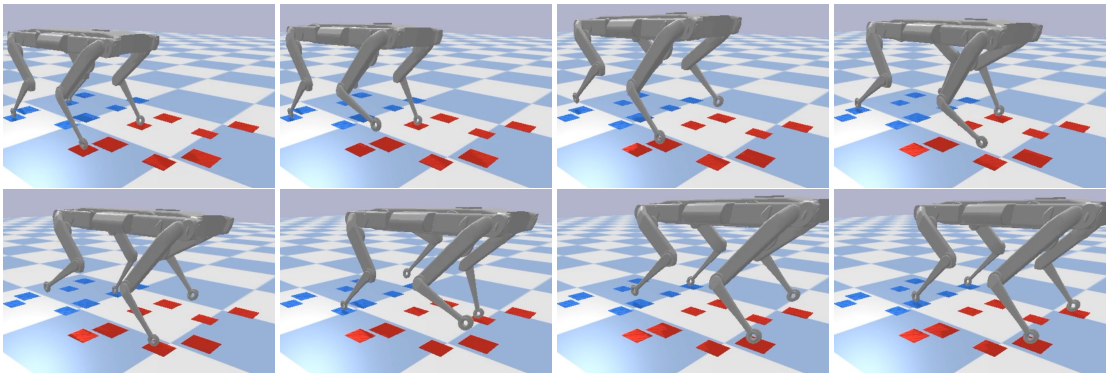
The feedforward torques are computed using the Recursive Newton-Euler Algorithm (RNEA) [30]. The joint position and velocity feedback gains are set to $\boldsymbol{K}_p = 2.\mathbb{I}_{n_j \times n_j}$ and $\boldsymbol{K}_d = 0.15.\mathbb{I}_{n_j \times n_j}$ respectively. The superscript * represents the optimized quantities coming from MPC. For the trot motion, NMPC and HNMPC failed to complete the motion by breaking contact in the second step as shown in Fig. 5.3a. On the contrary, SNMPC manages to complete the motion successfully until the end (see Fig. 5.3b). We tested the effect of leg inertia for an agile bounding motion, where NMPC and HNMPC failed again during the second bounding step (see Fig. 5.4a), while SNMPC completed the motion as shown in Fig. 5.4b (check the submission video).

5.3 Conclusions

In this work, we tackled the problem of kino-dynamic stochastic trajectory optimization subject to additive uncertainties on the dynamics and contact location chance constraints. We designed contact location safety constraints by computing upper bounds (back-offs) that take into account the linearized propagated uncertainties along the planning horizon assuming a Gaussian distribution of those uncertainties. The final solution is an approximate solution of the original SNMPC problem with a real-time iteration scheme. We compared the robustness of SNMPC against NMPC by running 1000 Monte-Carlo kino-dynamic simulations for agile



(a) Failed trot motion using NMPC.

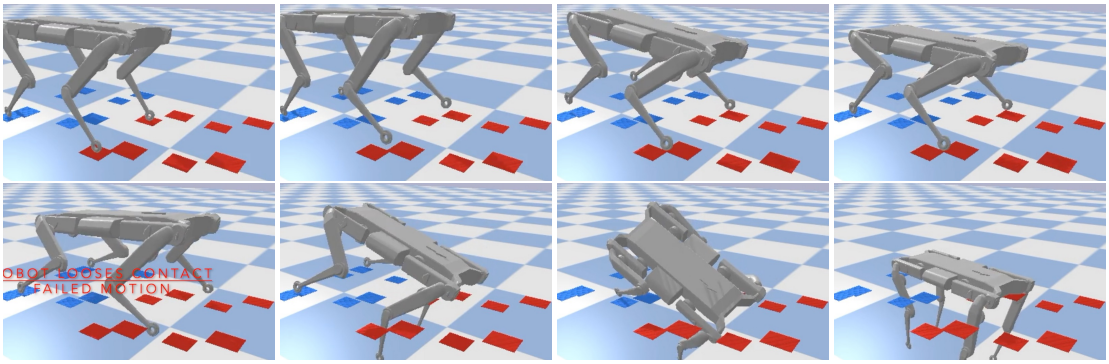


(b) Successful trot motion using SNMPC.

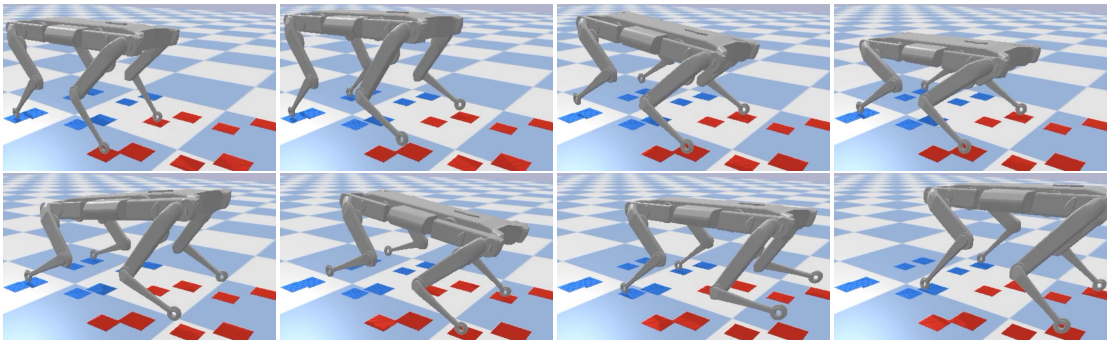
Figure 5.3: Comparison of whole-body trotting motion on non-coplanar stepping stones using NMPC and SNMPC.

trotting and bounding motions for the quadruped robot Solo on a challenging non-coplanar environment with small stepping stones as well as whole-body simulations. SNMPC completed all the motions successfully without violating the contact locations constraints, while NMPC violated them 48.3% of the time. Moreover, we ran whole-body simulations in Pybullet to study the effects of mismatch between the kino-dynamic and whole-body model; SNMPC was able to complete both motions successfully, while NMPC failed in both cases showing the benefit of SNMPC over deterministic planning in safety-critical scenarios.

Finally, we also compared the robustness of SNMPC gainst HNMPC. Since the robustness of SNMPC is induced by designing proper back-offs, then one might think why not design such safety margins heuristically by shrinking the constraint set by hand? We argue that although this approach might work in practice for some cases, it does not provide an automatic procedure of designing such safety margins leaving this to a process of trial and error. For instance, what should be the proper safety margin for different agile motion plans without degrading



(a) Failed bounding motion using NMPC.



(b) Successful bounding motion using SNMPC.

Figure 5.4: Comparison of whole-body bounding motion on non-coplanar stepping stones using NMPC and SNMPC.

performance? As shown in our empirical results, using the same heuristic safety margin for both trotting and bounding motions, yielded different safety rates of successful motions. Moreover, this heuristic-based approach does not relate the magnitude of the back-off design with the uncertainty statistics that might be available from previously collected data about the system in simulation or on the real robot. On the contrary, SNMPC methodologically addresses those issues by computing such bounds automatically, which vary at each point in time-based on expected uncertainty propagation along the horizon, the time-varying closed-loop feedback gain, and the desired probability of satisfying such constraints (4.7a).

One limitation of the current work is that it does not take into account contact mode uncertainties, which are of a combinatorial nature. We would like to explore tractable SNMPC formulations that take into account contact time uncertainties induced by uncertainties in the discrete contact modes, which is beneficial for sequential manipulation and locomotion tasks. Moreover, we intend to test the current SNMPC scheme on real robot experiments in future work.

Chapter 6

Summary and Conclusions

In the previous chapters, we introduced stochastic predictive control approaches for trajectory optimization of legged robots subject to state and control chance constraints under additive and parametric uncertainties.

In chapter 3, we dealt with the convex case of the simplified LIPM subject to individual (hyper-plane) collision avoidance chance constraints. By exploiting, the Gaussian assumption of uncertainties description and the model linearity, we were able to propagate exactly the expected error dynamics along the control horizon, and design time-varying constraint back-offs for constraint-tightening offline. The resulting reformulated SOC is a QP and has the same complexity as a convex nominal MPC online. Furthermore, we compared this SMPC algorithm against tube-based RMPC and nominal MPC applied to a simulated humanoid robot walking in a safety-critical situation while applying persistent stochastic disturbances as the robot passes through a narrow hallway. Our comparison focused on the trade-off between robustness (constraint satisfaction) and optimality (performance). Our tests show that, while SMPC does not provide hard guarantees on constraint satisfaction, in practice we did not observe any constraint violation with a sufficiently low expected probability of constraint violation of ($\leq$ 1%). This comes with the advantage of less conservative control, i.e. it results in better performance as measured by the cost function. We also highlighted the fundamental difference between RMPC and SMPC in terms of worst-case robustness, showing that SMPC sets attenuate maximum disturbance sets that shrink contractively as time grows. Loosely speaking, SMPC can be thought as a special kind of RMPC that considers shrinking disturbance sets along the horizon.

In chapter 4, we extended the stochastic OCP policy to the non-convex case of the centroidal momentum dynamics subject to additive and parametric fixed contact-location stochastic uncertainties and friction pyramid joint chance constraints. Using the same assumption of Gaussian uncertainties, we computed a linearization-based covariance propagation around the mean of the current SQP iteration to

compute a deterministic reformulation of the stochastic OCP. Furthermore, we presented a robust whole-body trajectory optimization framework that alternates between stochastic centroidal trajectory optimization and whole-body trajectory optimization for generating feasible robust whole-body motions. Although this stochastic OCP approach does not require additional optimization variables w.r.t. its deterministic counterpart, the computational complexity is relatively higher due to uncertainty propagation and additional tensor derivatives required for solving the linearized joint chance constraints. This framework was tested on a simulated quadruped on challenging unexpected terrains and different disturbance force disturbances on the robot. We showed that stochastic trajectory optimization was able to perform all motions more safely than nominal trajectory optimization with better performance.

The final result was presented in chapter 5. We presented an approximate real-time SNMPC algorithm for robust whole-body trajectory optimization of legged robots subject to additive uncertainties. Contrary to the previous work in chapter 4 on stochastic centroidal momentum trajectory optimization, we performed one whole-body trajectory optimization by optimizing for both the centroidal dynamics and the full robot kinematics (kino-dynamics), which allows us to model uncertainties on the optimized contact locations in a receding horizon fashion rather than on fixed contact locations with fixed parametric contact location uncertainties. For computational tractability, we performed a real-time SQP-type iteration by approximating the jacobian of the reformulated deterministic contact-location joint chance constraints. We ran extensive simulation robustness tests for agile quadrupedal motions subject to persistent force disturbances on the kino-dynamic model as well as whole-body simulations to account for the model mismatch between the kino-dynamic and whole-body model. Our results showed that SNMPC was able to perform all motions successfully till the end, while NMPC failed most of the time by violating the contact-location constraints. Finally, we discussed the safety margins introduced by the backed-off chance constraints against HNMPC where the safety margins are tuned by hand as a process of trial and error. Although this approach is simpler to implement and can work for some cases in practice, we argue that it can be tedious to tune for every optimized motion, and it does not relate the magnitude of back-off design with the uncertainty statistics that might be available from previously collected data about the system in simulation or on the real robot. On the contrary, SNMPC methodologically addresses those issues by computing such bounds automatically, which vary at each point in time based on expected uncertainty propagation along the horizon, the time-varying closed-loop feedback gain, and the desired probability of satisfying the chance constraints. We highlight that the final SNMPC algorithm has almost the same computational complexity as NMPC and HNMPC with negligible extra computation for the

analytical uncertainty propagation of the constraint back-off design at every MPC cycle.

6.1 Limitations

This thesis tackled some of the drawbacks of the state-of-the-art methods in robust trajectory optimization of legged robots by proposing for the first time to solve the problem using chance-constrained stochastic predictive control algorithms. However, we highlight that the presented algorithms are far from perfect, and there is room for improving the presented algorithms. We present some of those improvements in the next subsections.

6.1.1 Automatic Chance-Constraint Risk Allocation

In all of the presented algorithms, we assumed a fixed probability level for violating the chance-constraint for both the individual chance-constraints case in chapter 3, and an equally distributed risk allocation for the joint-chance constraints case between overall hyper-planes forming the constraint set in chapter 4, and chapter 5. Although this approach is rather simple and allows the same computational complexity as the deterministic case, it can be a bit conservative since classically one would like to allocate a larger back-off bound/safety margin on active constraints over non-active ones. Furthermore, we assumed fixed stabilizing dead-beat/LQR gains for the state feedback control policy. A better approach is to optimize as well for those parameters inside the stochastic OCP for better performance. This comes at a cost of increasing the number of optimization variables and increased computational complexity when solved using bi-level mathematical programs [81, 5, 110] for the linear SMPC or using an interior-point method as in [60]. Extending those approaches to SNMPC can be a good step to enhance the performance of the current algorithms.

6.1.2 Uncertainty Quantification

Throughout this work, we assumed that the additive and parametric uncertainties descriptions on the model dynamics are modeled as known Gaussian distributions. Although this allows the propagation of uncertainties in an exact analytical fashion in the linear SMPC case, the accuracy of the approximated linearization-based covariance propagation might not be accurate for long horizons in the nonlinear case, especially for offline trajectory optimization. One way to tackle this is by designing an ad-hoc uncertainty quantification module that learns the statistics of the underlying uncertainty distributions online for SNMPC. This

was explored in [44] applied to autonomous car racing using Gaussian Processes (GPs) for learning residual uncertainties in the process model. Other approaches like Pandala et al. learned the vertices of an additive polytopic set for linear RMPC offline in simulation for a quadruped walking on a variety of terrains [83]. Although this approach can learn such distributions offline, it can fall short on out-of-distribution uncertainties in the real world (a.k.a sim-to-real problem), which can hinder the safety and performance of the robot.

6.1.3 Contact Mode Uncertainties

Throughout the presented work, we explored dealing with additive and parametric uncertainties on the continuous part of the model dynamics for both linear and nonlinear cases. Another crucial uncertainty for robots in contact-rich scenarios is dealing with structural uncertainties in the discrete contact modes, which was not considered in the presented algorithms. This is crucial for long-horizon hybrid tasks such as locomotion and manipulation. This problem is very challenging due to the multi-modal nature of making/breaking contact with the environment along the horizon, which is computationally expensive. Coming up with tractable formulations for contact mode uncertainties inside SNMPC is essential for safe trajectory optimization. Combining ideas from scenario optimization and multi-stage stochastic programming is worth exploring [94].

6.2 Future Work

6.2.1 Inference for Control

Another direction to overcome the intractability of the stochastic OCP is by reformulating it as an inference problem by approximating a posterior distribution for sampling control sequences [106, 107]. This gives rise to a category of controllers known as Path Integral Controllers (PICs) [47, 116]. One of the challenges of such controllers is coming up with an efficient importance sampling procedure for selecting elite samples to approximate the control distribution. It was shown by Thijssen et al. [105] that the optimal feedback controller is indeed the optimal sampler by providing a zero-variance estimate. Another challenge is to incorporate constraints in the formulation of PIC. Recently, Carius et. al [14] applied PIC on real quadrupedal robot experiments under equality constraints. Although a big engineering effort was involved in the parallelization of a nominal MPC controller pipeline along with a learning module for selecting elite samples, it can be a promising direction given the recent advancement in computational power.

6.2.2 Dual Control/ Reinforcement Learning (RL)

In this thesis, we assumed perfect state knowledge. However, the two tasks of state estimation and control are closely intertwined; meaning that the control actions influence not only the system states but also the uncertainty associated with the states. Consider the case of a walking robot navigating with the camera through the world; what the robot's camera sees, affects the actions taken by the robot like where to make contact with the environment, and vice versa, the robot's actions affect how the camera is oriented. This duality in the perception-action loop was first studied by Feldbaum [31] gave rise to dual control theory, and has roots in operational research known as the Multi-Armed Bandit Problem [48]. The goal of dual controllers is to maintain an optimal balance (in the sense of the principle of optimality [2]) between the probing effect of reducing the uncertainty of the unknown parameters while improving the system's performance [73]. This trade-off is known as exploration-exploitation in the reinforcement learning literature [102]. From an optimal control perspective, those types of problems are intractable when attempted to be solved with Dynamic Programming (DP) [2] due to the curse of dimensionality. To this end, tractable sub-optimal solutions using approximate dynamic programming might come to rescue [4, 73].

6.2.3 Beyond H-2 and H-infinity controllers

The work in this thesis, as well as the majority of works on stochastic optimal control attempts in legged robotics, are categorized $\mathcal{H}_2$ control, where the expected value of the cost is minimized under an assumption of stochastic noise. On the other hand, works on robust optimal control like minimax approaches are categorized under $\mathcal{H}_\infty$ control which deals with worst-cost under adversarial disturbance. Committing to a specific control strategy can fall short when the disturbance does not follow the specific measure the policy is designed for in practice, leading to sub-optimal trajectories. To this end, designing adaptive controllers that can adjust their control strategy as they sequentially observe the disturbances irrespective of how the disturbance realizations are generated can be a better design goal for more rational controllers. A new promising step towards achieving this objective is minimizing *regret*. Regret minimization penalizes the loss suffered by a learner relative to the optimal policy in hindsight, and has its roots in statistical learning [92]. The goal of regret-based optimal control is to minimize the gap between a casual controller, which has access only to the current and previous history of the state against an offline non-casual controller assuming perfect knowledge of the future disturbances [35]. Extending linear regret-based controllers [35] to the nonlinear cases in legged locomotion can be a more robust metric for out-of-distribution scenario cases like impacts during walking.

Bibliography

[1] Joel A E Andersson, Joris Gillis, Greg Horn, James B Rawlings, and Moritz Diehl. CasADi – A software framework for nonlinear optimization and optimal control. *Mathematical Programming Computation*, 11(1):1–36, 2019.

[2] Richard Bellman. The theory of dynamic programming. *Bulletin of the American Mathematical Society*, 60(6):503–515, 1954.

[3] Alberto Bemporad and Manfred Morari. Robust model predictive control: A survey. In *Robustness in identification and control*, pages 207–226. Springer, 2007.

[4] Dimitri Bertsekas. *Dynamic programming and optimal control: Volume I*, volume 4. Athena scientific, 2012.

[5] Lars Blackmore and Masahiro Ono. Convex chance constrained predictive control without sampling. In *AIAA guidance, navigation, and control conference*, page 5876, 2009.

[6] Franco Blanchini and Stefano Miani. *Set-theoretic methods in control*. Springer, 2008.

[7] Néstor Bohórquez and Pierre-Brice Wieber. Adaptive step duration in biped walking: a robust approach to nonlinear constraints. In *2017 IEEE-RAS 17th International Conference on Humanoid Robotics (Humanoids)*, pages 724–729. IEEE, 2017.

[8] Riccardo Bonalli, Abhishek Cauligi, Andrew Bylard, and Marco Pavone. Gusto: Guaranteed sequential trajectory optimization via sequential convex programming. In *2019 International Conference on Robotics and Automation (ICRA)*, pages 6741–6747, 2019.

[9] James Bradbury, Roy Frostig, Peter Hawkins, Matthew James Johnson, Chris Leary, Dougal Maclaurin, George Necula, Adam Paszke, Jake VanderPlas, Skye Wanderman-Milne, and Qiao Zhang. JAX: composable transformations of Python+NumPy programs.

[10] Camille Brasseur, Alexander Sherikov, Cyrille Collette, Dimitar Dimitrov, and Pierre-Brice Wieber. A robust linear mpc approach to online generation of 3d biped walking motion. In *2015 IEEE-RAS 15th International Conference on Humanoid Robots (Humanoids)*, pages 595–601. IEEE, 2015.

[11] Rohan Budhiraja, Justin Carpentier, and Nicolas Mansard. Dynamics consensus between centroidal and whole-body models for locomotion of legged robots. In *2019 International Conference on Robotics and Automation (ICRA)*, pages 6727–6733, 2019.

[12] Mark Cannon. Model predictive control. *University of Oxford, Hilary Term*, 2016.

[13] Mark Cannon, Basil Kouvaritakis, Saša V Raković, and Qifeng Cheng. Stochastic tubes in model predictive control with probabilistic constraints. *IEEE Transactions on Automatic Control*, 56(1):194–200, 2010.

[14] Jan Carius, René Ranftl, Farbod Farshidian, and Marco Hutter. Constrained stochastic optimal control with learned importance sampling: A path integral approach. *The International Journal of Robotics Research*, 41(2):189–209, 2022.

[15] Justin Carpentier and Nicolas Mansard. Analytical derivatives of rigid body dynamics algorithms. In *Robotics: Science and Systems*, 2018.

[16] Justin Carpentier, Guilhem Saurel, Gabriele Buondonno, Joseph Mirabel, Florent Lamiraux, Olivier Stasse, and Nicolas Mansard. The pinocchio c++ library – a fast and flexible implementation of rigid body dynamics algorithms and their analytical derivatives. In *IEEE International Symposium on System Integrations (SII)*, 2019.

[17] Justin Carpentier and Pierre-Brice Wieber. Recent progress in legged robots locomotion control. *Current Robotics Reports*, 2(3):231–238, 2021.

[18] Yu-Ming Chen, Jianshu Hu, and Michael Posa. Beyond inverted pendulums: Task-optimal simple models of legged locomotion. *arXiv preprint arXiv:2301.02075*, 2023.

[19] L. Chisci, J.A. Rossiter, and G. Zappa. Systems with persistent disturbances: predictive control with restricted constraints. *Automatica*, 37(7):1019 – 1028, 2001.

[20] Matteo Ciocca, Pierre-Brice Wieber, and Thierry Fraichard. Effect of planning period on mpc-based navigation for a biped robot in a crowd. In

2019 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), pages 491–496, 2019.

[21] Erwin Coumans and Yunfei Bai. Pybullet, a python module for physics simulation for games, robotics and machine learning. http://pybullet.org, 2016–2021.

[22] Hongkai Dai and Russ Tedrake. Planning robust walking motion on uneven terrain via convex optimization. In *2016 IEEE-RAS 16th International Conference on Humanoid Robots (Humanoids)*, pages 579–586. IEEE, 2016.

[23] Hongkai Dai, Andrés Valenzuela, and Russ Tedrake. Whole-body motion planning with centroidal dynamics and full kinematics. In *2014 IEEE-RAS International Conference on Humanoid Robots*, pages 295–302, 2014.

[24] Elham Daneshmand, Majid Khadiv, Felix Grimminger, and Ludovic Righetti. Variable horizon mpc with swing foot dynamics for bipedal walking control. *IEEE Robotics and Automation Letters*, 6(2):2349–2356, 2021.

[25] Ewen Dantec, Maximilien Naveau, Pierre Fernbach, Nahuel Villa, Guilhem Saurel, Olivier Stasse, Michel Taix, and Nicolas Mansard. Whole-body model predictive control for biped locomotion on a torque-controlled humanoid robot. In *2022 IEEE-RAS 21st International Conference on Humanoid Robots (Humanoids)*, pages 638–644, 2022.

[26] Andrea Del Prete and Nicolas Mansard. Robustness to joint-torque-tracking errors in task-space inverse dynamics. *IEEE transactions on Robotics*, 32(5):1091–1105, 2016.

[27] Moritz Diehl, Hans Georg Bock, and Johannes P Schlöder. A real-time iteration scheme for nonlinear optimization in optimal feedback control. *SIAM Journal on control and optimization*, 43(5):1714–1736, 2005.

[28] Luke Drnach, John Z Zhang, and Ye Zhao. Mediating between contact feasibility and robustness of trajectory optimization through chance complementarity constraints. *Frontiers in Robotics and AI*, 8, 2021.

[29] Marcello Farina, Luca Giulioni, and Riccardo Scattolini. Stochastic linear model predictive control with chance constraints – a review. *Journal of Process Control*, 44:53 – 67, 2016.

[30] Roy Featherstone. *Rigid body dynamics algorithms*. Springer, 2014.

[31] ´AÁ Feldbâum. Dual control theory problems. *IFAC Proceedings Volumes*, 1(2):541–550, 1963.

[32] Ahmad Gazar, Majid Khadiv, Andrea Del Prete, and Ludovic Righetti. Multi-contact stochastic predictive control for legged robots with contact locations uncertainty. *arXiv preprint arXiv:2309.04469*, 2023.

[33] Ahmad Gazar, Majid Khadiv, Sébastien Kleff, Andrea Del Prete, and Ludovic Righetti. Nonlinear stochastic trajectory optimization for centroidal momentum motion generation of legged robots. In *Robotics Research*, pages 420–435, 2023.

[34] Ahmad Gazar, Majid Khadiv, Andrea Del Prete, and Ludovic Righetti. Stochastic and robust mpc for bipedal locomotion: A comparative study on robustness and performance. In *2020 IEEE-RAS 20th International Conference on Humanoid Robots (Humanoids)*, pages 61–68, 2021.

[35] Gautam Goel and Babak Hassibi. Regret-optimal estimation and control. *IEEE Transactions on Automatic Control*, 68(5):3041–3053, 2023.

[36] Ruben Grandia, Farbod Farshidian, René Ranftl, and Marco Hutter. Feedback mpc for torque-controlled legged robots. In *2019 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 4730–4737, 2019.

[37] Felix Grimminger, Avadesh Meduri, Majid Khadiv, Julian Viereck, Manuel Wüthrich, Maximilien Naveau, Vincent Berenz, Steve Heim, Felix Widmaier, Thomas Flayols, et al. An open torque-controlled modular robot architecture for legged locomotion research. *IEEE Robotics and Automation Letters*, 5(2), 2020.

[38] Bilal Hammoud, Armand Jordana, and Ludovic Righetti. irisc: Iterative risk sensitive control for nonlinear systems with imperfect observations. In *2022 American Control Conference (ACC)*, pages 3550–3557, 2022.

[39] Bilal Hammoud, Majid Khadiv, and Ludovic Righetti. Impedance optimization for uncertain contact interactions through risk sensitive optimal control. *IEEE Robotics and Automation Letters*, 6(3):4766–4773, 2021.

[40] Tor Aksel N. Heirung, Joel A. Paulson, Jared O'Leary, and Ali Mesbah. Stochastic model predictive control — how does it work? *Computers & Chemical Engineering*, 114:158 – 170, 2018. FOCAPO/CPC 2017.

[41] Andrei Herdt, Holger Diedam, Pierre-Brice Wieber, Dimitar Dimitrov, Katja Mombaur, and Moritz Diehl. Online walking motion generation with automatic foot step placement. *Advanced Robotics*, 24:719–737, 04 2010.

[42] Alexander Herzog, Stefan Schaal, and Ludovic Righetti. Structured contact force optimization for kino-dynamic motion generation. In *2016 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 2703–2710, 2016.

[43] Lukas Hewing, Andrea Carron, Kim P Wabersich, and Melanie N Zeilinger. On a correspondence between probabilistic and robust invariant sets for linear systems. In *2018 European Control Conference (ECC)*, pages 1642–1647. IEEE, 2018.

[44] Lukas Hewing, Alexander Liniger, and Melanie N Zeilinger. Cautious nmpc with gaussian process dynamics for autonomous miniature race cars. In *2018 European Control Conference (ECC)*, pages 1341–1348. IEEE, 2018.

[45] Lukas Hewing and Melanie N Zeilinger. Stochastic model predictive control for linear systems using probabilistic reachable sets. In *2018 IEEE Conference on Decision and Control (CDC)*, pages 5182–5188. IEEE, 2018.

[46] Armand Jordana, Bilal Hammoud, Justin Carpentier, and Ludovic Righetti. Stagewise newton method for dynamic game control with imperfect state observation. *IEEE Control Systems Letters*, 6:3241–3246, 2022.

[47] Hilbert J Kappen. Path integrals and symmetry breaking for optimal control theory. *Journal of statistical mechanics: theory and experiment*, 2005(11):P11011, 2005.

[48] Michael N Katehakis and Arthur F Veinott Jr. The multi-armed bandit problem: decomposition and computation. *Mathematics of Operations Research*, 12(2):262–268, 1987.

[49] Majid Khadiv, Alexander Herzog, S. Ali. A. Moosavian, and Ludovic Righetti. Walking control based on step timing adaptation. *IEEE Transactions on Robotics*, 36(3):629–643, 2020.

[50] Shahram Khorshidi, Ahmad Gazar, Nicholas Rotella, Maximilien Naveau, Ludovic Righetti, Maren Bennewitz, and Majid Khadiv. On the use of torque measurement in centroidal state estimation. In *2023 IEEE International Conference on Robotics and Automation (ICRA)*, pages 9931–9937, 2023.

[51] Sebastien Kleff, Avadesh Meduri, Rohan Budhiraja, Nicolas Mansard, and Ludovic Righetti. High-frequency nonlinear model predictive control of a manipulator. In *2021 IEEE International Conference on Robotics and Automation (ICRA)*, pages 7330–7336, 2021.

[52] Twan Koolen, Tomas De Boer, John Rebula, Ambarish Goswami, and Jerry Pratt. Capturability-based analysis and control of legged locomotion, part 1: Theory and application to three simple gait models. *The international journal of robotics research*, 31(9):1094–1113, 2012.

[53] Basil Kouvaritakis and Mark Cannon. Stochastic model predictive control., 2015.

[54] Harold Joseph Kushner. *Introduction to stochastic control*. Holt, Rinehart and Winston New York, 1971.

[55] Huibert Kwakernaak and Raphael Sivan. *Linear optimal control systems*, volume 1. Wiley-interscience New York, 1972.

[56] J. Köehler, M. A. Müller, and F. Allgöwer. A nonlinear model predictive control framework using reference generic terminal ingredients. *IEEE Transactions on Automatic Control*, pages 1–1, 2019.

[57] Thomas Lew, Riccardo Bonalli, and Marco Pavone. Chance-constrained sequential convex programming for robust trajectory optimization. In *2020 European Control Conference (ECC)*, 2020.

[58] He Li, Robert J Frei, and Patrick M Wensing. Model hierarchy predictive control of robotic systems. *IEEE Robotics and Automation Letters*, 6(2):3373–3380, 2021.

[59] M. Lorenzen, F. Dabbene, R. Tempo, and F. Allgöwer. Constraint-tightening and stability in stochastic model predictive control. *IEEE Transactions on Automatic Control*, 62(7):3165–3177, July 2017.

[60] Yudong Ma, Sergey Vichik, and Francesco Borrelli. Fast stochastic mpc with optimal risk allocation applied to building control systems. In *2012 IEEE 51st IEEE Conference on Decision and Control (CDC)*, pages 7559–7564, 2012.

[61] Anirudha Majumdar and Marco Pavone. How should a robot assess risk? towards an axiomatic theory of risk in robotics. In Nancy M. Amato, Greg Hager, Shawna Thomas, and Miguel Torres-Torriti, editors, *Robotics Research*, pages 75–84, Cham, 2020. Springer International Publishing.

[62] Anirudha Majumdar and Russ Tedrake. Funnel libraries for real-time robust feedback motion planning. *The International Journal of Robotics Research*, 36(8):947–982, 2017.

[63] Zachary Manchester and Scott Kuindersma. Dirtrel: Robust nonlinear direct transcription with ellipsoidal disturbances and lqr feedback. *Robotics, Sciences and Systems (RSS)*, 2017.

[64] Harry M. Markowitz. *Portfolio Selection: Efficient Diversification of Investments*. Yale University Press, 1959.

[65] Carlos Mastalli, Rohan Budhiraja, Wolfgang Merkt, Guilhem Saurel, Bilal Hammoud, Maximilien Naveau, Justin Carpentier, Ludovic Righetti, Sethu Vijayakumar, and Nicolas Mansard. Crocoddyl: An efficient and versatile framework for multi-contact optimal control. In *2020 IEEE International Conference on Robotics and Automation (ICRA)*, pages 2536–2542, 2020.

[66] Carlos Mastalli, Wolfgang Merkt, Guiyang Xin, Jaehyun Shim, Michael Mistry, Ioannis Havoutis, and Sethu Vijayakumar. Agile maneuvers in legged robots: a predictive control approach. *arXiv preprint arXiv:2203.07554*, 2022.

[67] D. Q. Mayne and W. Langson. Robustifying model predictive control of constrained linear systems. *Electronics Letters*, 37(23):1422–1423, 2001.

[68] David Q Mayne. Differential dynamic programming–a unified approach to the optimization of dynamic systems. In *Control and dynamic systems*, volume 10, pages 179–254. Elsevier, 1973.

[69] D.Q. Mayne. Robust and stochastic mpc: Are we going in the right direction? *IFAC-PapersOnLine*, 48(23):1 – 8, 2015. 5th IFAC Conference on Nonlinear Model Predictive Control NMPC 2015.

[70] D.Q. Mayne, J.B. Rawlings, C.V. Rao, and P.O.M. Scokaert. Constrained model predictive control: Stability and optimality. *Automatica*, 36(6):789 – 814, 2000.

[71] D.Q. Mayne, M.M. Seron, and S.V. Raković. Robust model predictive control of constrained linear systems with bounded disturbances. *Automatica*, 41(2):219 – 224, 2005.

[72] Avadesh Meduri, Paarth Shah, Julian Viereck, Majid Khadiv, Ioannis Havoutis, and Ludovic Righetti. Biconmp: A nonlinear model predictive control framework for whole body motion planning. *IEEE Transactions on Robotics*, 39(2):905–922, 2023.

[73] Ali Mesbah. Stochastic model predictive control with active uncertainty learning: A survey on dual control. *Annual Reviews in Control*, 45:107–117, 2018.

[74] Igor Mordatch, Kendall Lowrey, and Emanuel Todorov. Ensemble-cio: Full-body dynamic motion planning that transfers to physical humanoids. In *2015 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 5307–5314, 2015.

[75] Igor Mordatch, Emanuel Todorov, and Zoran Popović. Discovery of complex behaviors through contact-invariant optimization. *ACM Trans. Graph.*, 31(4), jul 2012.

[76] J. Morimoto, G. Zeglin, and C.G. Atkeson. Minimax differential dynamic programming: application to a biped walking robot. In *Proceedings 2003 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS 2003) (Cat. No.03CH37453)*, 2003.

[77] Yashwanth Kumar Nakka and Soon-Jo Chung. Trajectory optimization for chance-constrained nonlinear stochastic systems. In *2019 IEEE 58th Conference on Decision and Control (CDC)*, pages 3811–3818, 2019.

[78] Arkadi Nemirovski and Alexander Shapiro. Convex approximations of chance constrained programs. *SIAM Journal on Optimization*, 17(4):969–996, 2007.

[79] Jorge Nocedal and Stephen J Wright. *Numerical optimization*. Springer, 1999.

[80] Masahiro Ono, Lars Blackmore, and Brian C. Williams. Chance constrained finite horizon optimal control with nonconvex constraints. In *Proceedings of the 2010 American Control Conference*, pages 1145–1152, 2010.

[81] Masahiro Ono and Brian C. Williams. Iterative risk allocation: A new approach to robust model predictive control with a joint chance constraint. In *2008 47th IEEE Conference on Decision and Control*, pages 3427–3432, 2008.

[82] David E. Orin and Ambarish Goswami. Centroidal momentum matrix of a humanoid robot: Structure and properties. In *2008 IEEE/RSJ International Conference on Intelligent Robots and Systems*, pages 653–659, 2008.

[83] Abhishek Pandala, Randall T. Fawcett, Ugo Rosolia, Aaron D. Ames, and Kaveh Akbari Hamed. Robust predictive control for quadrupedal locomotion: Learning to close the gap between reduced- and full-order models. *IEEE Robotics and Automation Letters*, 7(3):6622–6629, 2022.

[84] J.A. Paulson, T.L.M. Santos, and A. Mesbah. Mixed stochastic-deterministic tube mpc for offset-free tracking in the presence of plant-model mismatch. *Journal of Process Control*, 83:102 – 120, 2019.

[85] Brian Plancher, Zachary Manchester, and Scott Kuindersma. Constrained unscented dynamic programming. In *2017 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 5674–5680, 2017.

[86] Michael Posa, Cecilia Cantu, and Russ Tedrake. A direct method for trajectory optimization of rigid bodies through contact. *The International Journal of Robotics Research*, 33(1):69–81, 2014.

[87] S. V. Rakovic, E. C. Kerrigan, K. I. Kouramas, and D. Q. Mayne. Invariant approximations of the minimal robust positively invariant set. *IEEE Transactions on Automatic Control*, 50(3):406–410, March 2005.

[88] Saša V. Raković and William Levine. *Handbook of Model Predictive Control*. 09 2018.

[89] J.B. Rawlings, D.Q. Mayne, and M. Diehl. *Model Predictive Control: Theory, Computation, and Design*. Nob Hill Publishing, 2017.

[90] Ludovic Righetti, Jonas Buchli, Michael Mistry, Mrinal Kalakrishnan, and Stefan Schaal. Optimal distribution of contact forces with inverse-dynamics control. *The International Journal of Robotics Research*, 32(3):280–298, 2013.

[91] T. L. M. Santos, A. D. Bonzanini, T. A. N. Heirung, and A. Mesbah. A constraint-tightening approach to nonlinear model predictive control with chance constraints for stochastic systems. In *2019 American Control Conference (ACC)*, pages 1641–1647, 2019.

[92] L. J. Savage. The theory of statistical decision. *Journal of the American Statistical Association*, 46(253):55–67, 1951.

[93] Andrey V. Savkin and Ian R. Petersen. Minimax optimal control of uncertain systems with structured uncertainty. *International Journal of Robust and Nonlinear Control*, 5(2):119–137, 1995.

[94] Georg Schildbach. *Scenario-based optimization for multi-stage stochastic decision problems*. PhD thesis, ETH Zurich, 2014.

[95] John Schulman, Yan Duan, Jonathan Ho, Alex Lee, Ibrahim Awwal, Henry Bradlow, Jia Pan, Sachin Patil, Ken Goldberg, and Pieter Abbeel. Motion planning with sequential convex optimization and convex collision checking. *The International Journal of Robotics Research*, 2014.

[96] P.O.M. Scokaert and D.Q. Mayne. Min-max feedback model predictive control for constrained linear systems. *IEEE Transactions on Automatic Control*, 43(8):1136–1142, 1998.

[97] Alexander Shapiro, Darinka Dentcheva, and Andrzej Ruszczynski. *Lectures on stochastic programming: modeling and theory*. SIAM, 2021.

[98] Alexander Sherikov, Dimitar Dimitrov, and Pierre-Brice Wieber. Whole body motion controller with long-term balance constraints. In *2014 IEEE-RAS International Conference on Humanoid Robots*, pages 444–450. IEEE, 2014.

[99] Herbert A Simon. Dynamic programming under uncertainty with a quadratic criterion function. *Econometrica, Journal of the Econometric Society*, pages 74–81, 1956.

[100] O. Stasse, T. Flayols, R. Budhiraja, K. Giraud-Esclasse, J. Carpentier, J. Mirabel, A. Del Prete, P. Soueres, N. Mansard, F. Lamiraux, J.-P. Laumond, L. Marchionni, H. Tome, and F. Ferro. TALOS: A new humanoid research platform targeted for industrial applications. In *IEEE-RAS International Conference on Humanoid Robots*, volume Part F1341, 2017.

[101] Hyung Ju Terry Suh, Tao Pang, and Russ Tedrake. Bundled gradients through contact via randomized smoothing. *IEEE Robotics and Automation Letters*, 7(2):4000–4007, 2022.

[102] Richard S Sutton and Andrew G Barto. *Reinforcement learning: An introduction*. MIT press, 2018.

[103] Y. Tassa and E. Todorov. Stochastic complementarity for local control of discontinuous dynamics. In *Proceedings of Robotics: Science and Systems*, June 2010.

[104] Yuval Tassa, Tom Erez, and Emanuel Todorov. Synthesis and stabilization of complex behaviors through online trajectory optimization. In *2012 IEEE/RSJ International Conference on Intelligent Robots and Systems*, pages 4906–4913, 2012.

[105] Sep Thijssen and HJ Kappen. Path integral control and state-dependent feedback. *Physical Review E*, 91(3):032104, 2015.

[106] Emanuel Todorov. General duality between optimal control and estimation. In *2008 47th IEEE Conference on Decision and Control*, pages 4286–4292, 2008.

[107] Marc Toussaint. Robot trajectory optimization using approximate inference. In *Proceedings of the 26th Annual International Conference on Machine Learning*, ICML '09, page 1049–1056, New York, NY, USA, 2009. Association for Computing Machinery.

[108] Robin Verschueren, Gianluca Frison, Dimitris Kouzoupis, Jonathan Frey, Niels van Duijkeren, Andrea Zanelli, Branimir Novoselnik, Thivaharan Albin, Rien Quirynen, and Moritz Diehl. acados – a modular open-source framework for fast embedded optimal control. *Mathematical Programming Computation*, 2021.

[109] N. A. Villa and P. Wieber. Model predictive control of biped walking with bounded uncertainties. In *2017 IEEE-RAS 17th International Conference on Humanoid Robotics (Humanoids)*, pages 836–841, 2017.

[110] Michael P. Vitus and Claire J. Tomlin. On feedback design and risk allocation in chance constrained control. In *2011 50th IEEE Conference on Decision and Control and European Control Conference*, pages 734–739, 2011.

[111] Jiayi Wang, Sanghyun Kim, Sethu Vijayakumar, and Steve Tonneau. Multi-fidelity receding horizon planning for multi-contact locomotion. In *2020 IEEE-RAS 20th International Conference on Humanoid Robots (Humanoids)*, pages 53–60, 2021.

[112] Patrick M Wensing, Michael Posa, Yue Hu, Adrien Escande, Nicolas Mansard, and Andrea Del Prete. Optimization-based control for dynamic legged robots. *arXiv preprint arXiv:2211.11644*, 2022.

[113] Pierre-brice Wieber. Trajectory free linear model predictive control for stable walking in the presence of strong perturbations. In *2006 6th IEEE-RAS International Conference on Humanoid Robots*, pages 137–142, 2006.

[114] Pierre-Brice Wieber, Russ Tedrake, and Scott Kuindersma. *Modeling and Control of Legged Robots*, pages 1203–1234. Springer International Publishing, 2016.

[115] Georg Wiedebach, Sylvain Bertrand, Tingfan Wu, Luca Fiorio, Stephen McCrory, Robert Griffin, Francesco Nori, and Jerry Pratt. Walking on partial footholds including line contacts with the humanoid robot atlas. In *2016 IEEE-RAS 16th International Conference on Humanoid Robots (Humanoids)*, pages 1312–1319. IEEE, 2016.

[116] Grady Williams, Andrew Aldrich, and Evangelos A Theodorou. Model predictive path integral control: From theory to parallel computation. *Journal of Guidance, Control, and Dynamics*, 40(2):344–357, 2017.

[117] Alexander W. Winkler, C. Dario Bellicoso, Marco Hutter, and Jonas Buchli. Gait and trajectory optimization for legged systems through phase-based end-effector parameterization. *IEEE Robotics and Automation Letters*, 3(3):1560–1567, 2018.

[118] Hans S Witsenhausen. Separation of estimation and control for discrete time systems. *Proceedings of the IEEE*, 59(11):1557–1566, 1971.

[119] Shaohang Xu, Lijun Zhu, Hai-Tao Zhang, and Chin Pang Ho. Robust convex model predictive control for quadruped locomotion under uncertainties. *IEEE Transactions on Robotics*, pages 1–18, 2023.

[120] Mohammad Hasan Yeganegi, Majid Khadiv, S. Ali A. Moosavian, Jia-Jie Zhu, Andrea Del Prete, and Ludovic Righetti. Robust humanoid locomotion using trajectory optimization and sample-efficient learning. In *2019 IEEE-RAS 19th International Conference on Humanoid Robots (Humanoids)*, pages 170–177, 2019.

[121] Hai Zhu and Javier Alonso-Mora. Chance-constrained collision avoidance for mavs in dynamic environments. *IEEE Robotics and Automation Letters*, 4(2):776–783, 2019.