Closed-Loop Belief Space Planning for Linear, Gaussian Systems

Michael P. Vitus
Department of Aeronautics and Astronautics
Stanford University
Email: vitus@stanford.edu

Claire J. Tomlin
Department of Electrical Engineering and Computer Sciences
University of California at Berkeley
Email: tomlin@eecs.berkeley.edu

Abstract— **This paper considers the problem of motion planning for linear, Gaussian systems, and extends existing chance constrained optimal control solutions [1], [2] by incorporating the closed-loop uncertainty of the system and by reducing the conservativeness in the constraints. Due to the imperfect knowledge of the system state caused by motion uncertainty and sensor noise, the constraints cannot be guaranteed to be satisfied and consequently must be considered probabilistically. In this work, they are formulated as convex constraints on a univariate Gaussian random variable, with the violation probability of all the constraints guaranteed to be below a threshold. This threshold is a tuning parameter which trades off the performance of the system and the conservativeness of the solution. In contrast to similar methods, the proposed work considers the specific estimator and controller used in the closed-loop system in order to directly characterize the *a priori* distribution of the closed-loop system state. Using this distribution, a convex optimization program is formulated to solve for the optimal solution for the closed-loop system. The performance of the algorithm is demonstrated through several examples.**

I. Introduction

With the increasing demand for autonomous vehicles, there is a pressing need for algorithms to plan trajectories through complex environments. Several examples of projects that require this technology are: search and rescue, reconnaissance, surveillance, and disaster response. The general motion planning problem involves generating a trajectory that accomplishes a defined goal while operating under specified system dynamics and constraints. There has been a large amount of work in path planning for deterministic systems, but not as much for stochastic systems; although they have been receiving a great deal of attention in recent years.

The world is full of disturbances whether it be in the form of external sources, modeling errors or sensing noise; therefore the exact system state is never truly known. Consequently, in order to maximize the probability of success, the planning problem must be performed in the space of probability distributions of the system state, defined as the belief space. For a stochastic system, however, planning in the belief space is not enough to guarantee success because there is always a small probability that a large disturbance will be experienced. Therefore, a trade-off must be made between the conservativeness of the plan and the performance of the system.

This work is supported in part by NASA (#NNA06CN22A) and ONR under MURI grant N00014-09-1-1051.

The problem of motion planning with motion noise, sensing noise and environment uncertainty has been studied in the past. Some previous planners [3], [4] account for the motion uncertainty of the system but do not account for the partial observability of the system state or the sensing uncertainty. Others [5], [6], [7] [8] have included both the motion and sensing uncertainty when planning paths through the environment, but they simplify the problem by assuming the maximum likelihood observation is received for all future time-steps. This approximation results in an inaccurate representation of the probability distribution of the state which can lead to a violation of the constraints on the system.

Another group of researchers formulated the problem as an optimal control problem with constraints on the system state referred to as *chance constraints*. In [9], they extended their previous work to handle non-Gaussian belief distributions by approximating it using a finite number of particles. This transforms the original stochastic control problem into a deterministic one that can be efficiently solved. This sampling approach, however, becomes intractable as the number of samples needed to fully represent the true belief state increases. The work by Blackmore et al. [1] uses the work presented in [10] to approximate the chance constraints using Boole's inequality which typically leads to a very small amount of over-conservativeness. They also used the idea of risk allocation introduced by [10] to distribute the risk of violating each chance constraint while still guaranteeing the specified level of safety. By using the risk allocation technique instead of assuming a constant amount of risk for each constraint, the performance of the overall system can be significantly increased.

The work by van Hessem [2] optimized over the feedback control laws and open-loop inputs while ensuring that the chance constraints on the overall system are satisfied. They used an ellipsoidal set bounding approach to convert the stochastic problem into a deterministic one but this leads to a conservative solution. Also, their results only hold in finite-horizon (open-loop) execution and not in receding horizon (closed-loop) [1].

The problem of planning with motion noise can alternatively be handled by robust control techniques [11], [12]. Rather than using chance constraints, these methods assume bounds on the unknown parameters which can be used to formulate worst case bounds on the system state. However, such an approach is conservative because it disregards the information that is often available about the distribution of the state uncertainty.

van den Berg et al. [13] characterized the probability distributions of a system based upon a linear quadratic regulator controller with Gaussian models of uncertainty. They proposed a two step planning process: (1) a set of candidate paths were generated without taking into account the uncertainty of the system, (2) they selected the best path based upon a criterion which incorporated the uncertainty of the state. Since the uncertainty wasn't taken into account in generating the paths, the paths generated were often suboptimal solutions.

This work is based on the material presented in [1], [2] which formulates the motion planning problem as a chance constrained optimal control problem. This work extends [1] by accounting for the closed-loop uncertainty directly and extends [2] by optimizing the risk allocation to reduce the conservativeness of the solution. The authors' previous work [14] is also used if the constraints form a non-convex region to decompose them into a set of convex constraints which can be searched over to find the optimal solution. The algorithm is demonstrated to be very fast and could be applied in real-time or used to re-plan on-the-fly.

The paper proceeds as follows. Section II describes the probabilistic problem formulation. Then, the explicit estimator and controller used in the closed-loop system is formally presented in Section III. In Section IV, the *a priori* uncertainty of the closed-loop system is derived which is used in evaluating the chance constraints of the optimization program presented in Section V. The final convex optimization program is presented in Section VI, and several examples are presented in Section VII which characterize the performance of the algorithm. The paper concludes with directions of future work.

II. Problem Formulation

Consider the following linear stochastic system defined by,

$$x_{k+1} = Ax_k + Bu_k + w_k,\ \forall k \in [0, N-1], \tag{1}$$

where $x_k \in \mathbb{R}^n$ is the system state, $w_k \in \mathbb{R}^n$ is the process noise and N is the time horizon. The initial state, x_0, is assumed to be a Gaussian random variable with mean $\bar{x}_0$ and covariance Σ_0 i.e., $x_0 \sim \mathcal{N}(\bar{x}_0, \Sigma_0)$. At each time step, a noisy measurement of the state is taken, defined by

$$y_k = Cx_k + v_k,\ \forall k \in [1, N], \tag{2}$$

where $y_k \in \mathbb{R}^p$ and $v_k \in \mathbb{R}^p$ are the measurement output and noise of the sensor at time k, respectively. The process and measurement noise have zero mean Gaussian distributions, $w_k \sim \mathcal{N}(0, \Sigma_w)$ and $v_k \sim \mathcal{N}(0, \Sigma_v)$. The process noise, measurement noise and initial state are assumed to be mutually independent. For notational convenience, the state and control inputs for all time-steps are concatenated to form, $\mathbb{X} = \begin{bmatrix} x_1^{\mathrm{T}} & \ldots & x_N^{\mathrm{T}} \end{bmatrix}^{\mathrm{T}}$ and $\mathbb{U} = \begin{bmatrix} u_0^{\mathrm{T}} & \ldots & u_{N-1}^{\mathrm{T}} \end{bmatrix}^{\mathrm{T}}$. The control inputs are required to be in a convex region denoted by F_U and the system state is restricted to be in a feasible region denoted by F_X. The optimization program that is being solved is shown in Program P2.1 which is a general belief space planning problem. The objective function, $f(\cdot)$, is assumed to be a convex function in $\mathbb{X}$ and $\mathbb{U}$.

Path Planning Problem

$$\begin{aligned} \text{minimize} \quad & \mathbf{E}\left(f(\mathbb{X}, \mathbb{U})\right) \\ \text{subject to} \quad & \\ & x_{k+1} = Ax_k + Bu_k + w_k,\ \forall k \in [0, N{-}1] \\ & y_k = Cx_k + v_k,\ \forall k \in [1, N] \\ & w_k \sim \mathcal{N}(0, \Sigma_w),\ \forall k \in [0, N{-}1] \\ & v_k \sim \mathcal{N}(0, \Sigma_v),\ \forall k \in [1, N] \\ & \mathbb{U} \in F_U \\ & \mathrm{P}(\mathbb{X} \notin F_X) \le \delta \end{aligned} \tag{P2.1}$$

The difficulty in solving the optimization program P2.1 is in evaluating the chance constraints: $\mathrm{P}(\mathbb{X} \notin F_X) \le \delta$. The complexity arises from characterizing the probability distribution of the system state at each time-step and evaluating the multivariate integrals over the feasible region. The closed-loop probability distribution is even more difficult since the properties of the estimator and controller need to be taken into account. Consequently, the specific estimator and controller used in the system need to be defined to characterize the closed-loop uncertainty of the system which will then be used to evaluate the chance constraints.

III. System Description

The problem formulation in Program P2.1 is independent of the type of estimator and controller that is used in the actual system. In this work, a Kalman filter is used as the estimator and the system is controlled via a linear quadratic trajectory controller.

A. Kalman Filter

For a linear, Gaussian system, the Kalman filter is the minimum mean square error estimator and is used to estimate the mean and uncertainty of the system state. Let $\hat{\Sigma}_{k|k-1}$ be the covariance matrix of the optimal estimate of x_k given the measurements $\{y_1, \ldots, y_{k-1}\}$ and let $\hat{\Sigma}_{k|k}$ be the covariance matrix of the optimal estimate of x_k given the measurements $\{y_1, \ldots, y_k\}$. The Kalman filter operates through the following recursion. The process update is,

$$\hat{x}_{k|k-1} = A\hat{x}_{k-1|k-1} + Bu_{k-1} \tag{3}$$

$$\hat{\Sigma}_{k|k-1} = A\hat{\Sigma}_{k-1|k-1}A^{\mathrm{T}} + \Sigma_w. \tag{4}$$

The measurement update is,

$$\hat{x}_{k|k} = \hat{x}_{k|k-1} + L_k\left(y_k - C\hat{x}_{k|k-1}\right) \tag{5}$$

$$\hat{\Sigma}_{k|k} = (I - L_kC)\,\hat{\Sigma}_{k|k-1} \tag{6}$$

where $I \in \mathbb{R}^{n \times n}$ is the identity matrix and

$$L_k = \hat{\Sigma}_{k|k-1}C^{\mathrm{T}}\left(C\hat{\Sigma}_{k|k-1}C^{\mathrm{T}} + \Sigma_v\right)^{-1}. \tag{7}$$

One of the key properties of the Kalman filter is that the covariance, $\hat{\Sigma}_{k|k}$, can be pre-computed *before* any measurements are received which allows the uncertainty of the state to be utilized in the path planning process.

B. Linear Quadratic Trajectory Control

Given perfect state information, the linear quadratic controller minimizes the following cost function,

$$J = \underset{u_k, \forall k \in T_N}{\text{minimize}} \sum_{\tau=0}^{N} \left(x_\tau - x_\tau^d\right)^{\mathrm{T}} Q \left(x_\tau - x_\tau^d\right) + \sum_{\tau=0}^{N-1} u_\tau^{\mathrm{T}} R u_\tau, \tag{8}$$

where x_k^d is the desired trajectory at time-step k. It can be shown that the optimal input for the system in Eqn. (1) is an affine function of the current state given by,

$$u_k^* = K_k x_k + g_k \tag{9}$$

$$K_k = -\left(R + B^{\mathrm{T}} P_{k+1} B\right)^{-1} B^{\mathrm{T}} P_{k+1} A \tag{10}$$

$$g_k = -\left(R + B^{\mathrm{T}} P_{k+1} B\right)^{-1} B^{\mathrm{T}} q_{k+1} \tag{11}$$

where P_k and q_k are calculated via,

$$\begin{aligned} P_k = \; & Q + A^{\mathrm{T}} P_{k+1} A - \\ & A^{\mathrm{T}} P_{k+1} B \left(R + B^{\mathrm{T}} P_{k+1} B\right)^{-1} B^{\mathrm{T}} P_{k+1} A \end{aligned} \tag{12}$$

$$q_k = \left(A + B K_k\right)^{\mathrm{T}} q_{k+1} - Q x_k^d \tag{13}$$

starting from $P_N = Q$ and $q_N = -Q^{\mathrm{T}} x_N^d$.

C. Linear Quadratic Gaussian

Given that there is process and measurement noise, the exact value of the state x_k is not known. Therefore only the expectation of the cost function in Eqn. (8) can be minimized. Fortunately, the certainty equivalence principle has been used to show that the optimal controller for a linear, Gaussian system with quadratic cost uses the estimate of the state from a Kalman filter in Eqn. (9), i.e.,

$$u_k^* = K_k \hat{x}_{k|k} + g_k. \tag{14}$$

IV. *A PRIORI* DISTRIBUTION OF THE CLOSED-LOOP STATE AND CONTROL INPUT

Now that the estimator and controller have been defined, the probability distribution of the closed-loop system state, $\mathbb{X}$, can be formulated which is required to evaluate the chance constraints, $\mathrm{P}(\mathbb{X} \notin F_X) \leq \delta$.

Applying the methodology developed in [13], the closed-loop uncertainty of the system state can be characterized *a priori* before any measurements are received. Given the assumption of the Kalman filter estimator and linear quadratic trajectory tracking controller, the true system evolves according to,

$$x_{k+1} = A x_k + B(K_k \hat{x}_{k|k} + g_k) + w_k, \; \forall k \in [1, N], \tag{15}$$

and the estimate of the system state evolves according to,

$$\begin{aligned} \hat{x}_{k+1|k+1} = \; & A \hat{x}_{k|k} + B\left(K_k \hat{x}_{k|k} + g_k\right) + \\ & L_{k+1}\left(y_{k+1} - C \hat{x}_{k+1|k}\right). \end{aligned} \tag{16}$$

Substituting in for y_{k+1} and $\hat{x}_{k+1|k}$ from Eqns. (2) and (3) into Eqn. (16) yields,

$$\begin{aligned} \hat{x}_{k+1|k+1} = \; & A \hat{x}_{k|k} + B\left(K_k \hat{x}_{k|k} + g_k\right) + \\ & L_{k+1}\left[C\left(A x_k + B(K_k \hat{x}_{k|k} + g_k) + w_k\right) + \right. \\ & \left. v_{k+1} - C\left(A \hat{x}_{k|k} + B\left(K_k \hat{x}_{k|k} + g_k\right)\right)\right]. \end{aligned} \tag{17}$$

Finally, simplifying and collecting the common terms results in,

$$\begin{aligned} \hat{x}_{k+1|k+1} = \; & \left(A + B K_k - L_{k+1} C A\right) \hat{x}_{k|k} + L_{k+1} C A x_k + \\ & B g_k + L_{k+1} C w_k + L_{k+1} v_{k+1}. \end{aligned} \tag{18}$$

Combining x_{k+1} and $\hat{x}_{k+1|k+1}$ from Eqns. (15) and (18) into a single system results in the following time-varying system,

$$z_{k+1} = F_k z_k + \bar{B} g_k + G_k s_k \tag{19}$$

where

$$\begin{gathered} z_k = \begin{bmatrix} x_k \\ \hat{x}_{k|k} \end{bmatrix}, \quad s_k = \begin{bmatrix} w_k \\ v_{k+1} \end{bmatrix} \quad \bar{B} = \begin{bmatrix} B \\ B \end{bmatrix}, \\ F_k = \begin{bmatrix} A & B K_k \\ L_{k+1} C A & A + B K_k - L_{k+1} C A \end{bmatrix}, \\ G_k = \begin{bmatrix} I & 0 \\ L_{k+1} C & L_{k+1} \end{bmatrix} \end{gathered} \tag{20}$$

with $s_k \sim \mathcal{N}\left(0, \Sigma_s\right)$, $\Sigma_s = \operatorname{diag}\left(\Sigma_w, \Sigma_v\right)$ and diag places the matrices along the diagonal. The system starts from $z_0 = \begin{bmatrix} x_0^{\mathrm{T}} & \bar{x}_0^{\mathrm{T}} \end{bmatrix}^{\mathrm{T}}$. The mean, $\bar{z}_k$, and covariance, M_k, for the system defined by Eqn. (19) can now be determined,

$$\bar{z}_{k+1} = F_k \bar{z}_k + \bar{B} g_k \tag{21}$$

$$M_{k+1} = F_k M_k F_k^{\mathrm{T}} + G_k \Sigma_s G_k^{\mathrm{T}}, \tag{22}$$

starting from $\bar{z}_0 = \begin{bmatrix} \bar{x}_0 \\ \bar{x}_0 \end{bmatrix}$ and $M_0 = \begin{bmatrix} \Sigma_0 & 0 \\ 0 & 0 \end{bmatrix}$. Finally, the closed-loop state and control input evolves according to,

$$\begin{bmatrix} x_k \\ u_k \end{bmatrix} = \underbrace{\begin{bmatrix} I & 0 \\ 0 & K_k \end{bmatrix}}_{\Lambda_k} \begin{bmatrix} x_k \\ \hat{x}_{k|k} \end{bmatrix} + \begin{bmatrix} 0 \\ I \end{bmatrix} g_k. \tag{23}$$

Consequently, the a-priori closed-loop state and control input at time-step k, $\forall k \in [1, N-1]$, is distributed by,

$$\begin{bmatrix} x_k \\ u_k \end{bmatrix} \sim \mathcal{N}\left(\begin{bmatrix} T_k^{xx} \bar{x}_0 + T_k^{xg} \mathbf{g} \\ K_k (T_k^{xx} \bar{x}_0 + T_k^{xg} \mathbf{g}) + g_k \end{bmatrix}, \Lambda_k M_k \Lambda_k^{\mathrm{T}}\right) \tag{24}$$

where $\mathbf{g} = \begin{bmatrix} g_0^{\mathrm{T}} & \ldots & g_{N-1}^{\mathrm{T}} \end{bmatrix}^{\mathrm{T}}$. The k-th row of the matrix T^{xx} is,

$$T_k^{xx} = \prod_{\tau=0}^{k-1} (A + B K_\tau) \tag{25}$$

and the elements of the matrix T^{xg} are,

$$T_{ij}^{xg} = \begin{cases} 0 & \text{if } j > i \\ B & \text{if } j = i \\ (\prod_{\tau=j+1}^{i} (A + B K_\tau)) B & \text{otherwise} \end{cases} \tag{26}$$

Let $\mathbb{X}^d$ and $\mathbb{Q}$ be defined as: $\mathbb{X}^d = \begin{bmatrix} x_1^{d\,\mathrm{T}} & \ldots & x_N^{d\,\mathrm{T}} \end{bmatrix}^{\mathrm{T}}$ and $\mathbb{Q} = \begin{bmatrix} q_1^{\mathrm{T}} & \ldots & q_N^{\mathrm{T}} \end{bmatrix}^{\mathrm{T}}$. The constant input term in Eqn. (14) can be defined as a function of $\mathbb{Q}$, i.e., $\mathbf{g} = \Psi\, \mathbb{Q}$ where Ψ is a block diagonal matrix with entries $\Psi_{ii} = -\ (R + B^{\mathrm{T}} P_{i+1} B)^{-1} B^{\mathrm{T}}, \forall i \in [0, N-1]$. Also, the vector $\mathbf{q}$ can be written in terms of the desired trajectory, $\mathbb{Q} = \Phi \mathbb{X}^d$ where Φ is defined as $\forall i, j \in [0, N-1]$,

$$\Phi_{ij} = \begin{cases} 0 & \text{if } j < i \\ -Q^{\mathrm{T}} & \text{if } j = i \\ -(\prod_{\tau=i+1}^{j} (A + BK_\tau)^{\mathrm{T}}) Q^{\mathrm{T}} & \text{otherwise.} \end{cases} \tag{27}$$

Consequently, the mean of the closed-loop state is,

$$\bar{\mathbb{X}} = T^{xx} \bar{x}_0 + T^{xxd} \mathbb{X}^d \tag{28}$$

where $T^{xxd} = T^{xg} \Psi \Phi$. The mean of the control inputs is,

$$\bar{\mathbb{U}} = T^{ux_0} \bar{x}_0 + T^{uxd} \mathbb{X}^d \tag{29}$$

with $T^{ux_0} = \tilde{K} T^{xx} + \begin{bmatrix} K_0 \\ 0 \end{bmatrix}$, $T^{uxd} = \left(\tilde{K} T^{xx^d} + \Psi \Phi \right)$, and $\tilde{K}$ is a block matrix with $[K_1, \ldots, K_{N-1}]$ on the lower off-diagonal.

V. Chance Constraints

Using the probability distribution of the system state formulated in the previous section, the chance constraints, $\mathrm{P}(\mathbb{X} \notin F_X) \leq \delta$, in Program P2.1 can be evaluated efficiently once the multivariate integrals over the feasible region are simplified.

A. Univariate Gaussian Constraints

While the state chance constraints may first appear to be easy to evaluate, they require the integration of a multivariate Gaussian density which does not have an analytic solution. In [9], a sampling strategy was exploited to evaluate the integral, but this is computationally expensive because it requires a large number of samples for the required accuracy. Fortunately, [10], [1] showed that the multivariate integrals can be converted into univariate integrals which can be evaluated efficiently.

For simplicity, the feasible region F_X is assumed to be convex (although non-convex regions can be handled by convexifying the space and using branch and bound in an outer-loop). Given this assumption, the feasible region can be defined by a conjunction of N_{F_X} linear inequality constraints,

$$F_X \triangleq \bigcap_{i=1}^{N_{F_X}} \left\{ \mathbb{X} : a_i^{\mathrm{T}} \mathbb{X} \leq b_i \right\} \tag{30}$$

where $a_i \in \mathbb{R}^{nN}$ and $b_i \in \mathbb{R}$. By using Boole's inequality, the original chance constraint $\mathrm{P}(\mathbb{X} \notin F_X)$ can be conservatively approximated. Boole's inequality states that for a countable set of events $E_1, E_2, \ldots$, the probability that at least one event happens is no larger than the sum of the individual probabilities,

$$\mathrm{P}\left(\bigcup_i E_i \right) \leq \sum_i \mathrm{P}\left(E_i \right). \tag{31}$$

Consequently, from Eqn. (30) and Boole's inequality the probability of the state not being contained inside the feasible region is bounded by,

$$\begin{aligned} \mathrm{P}(\mathbb{X} \notin F_X) \quad &= \quad \mathrm{P}\left(\mathbb{X} \in \bigcup_{i=1}^{N_{F_X}} \left\{ \mathbb{X} : a_i^{\mathrm{T}} \mathbb{X} > b_i \right\} \right) \\ &\leq \quad \textstyle\sum_{i=1}^{N_{F_X}} \mathrm{P}(a_i^{\mathrm{T}} \mathbb{X} > b_i). \end{aligned} \tag{32}$$

From Eqns. (24) and (28) the closed-loop state is a Gaussian random variable defined by,

$$\mathbb{X} \sim \mathcal{N}\left(\bar{\mathbb{X}}, \mathbb{M} \right) \tag{33}$$

where $\mathbb{M} = \mathrm{diag}(\tilde{I}^{\mathrm{T}} M_1 \tilde{I}, \ldots, \tilde{I}^{\mathrm{T}} M_N \tilde{I})$ and $\tilde{I} = [I\ \mathbf{0}]^{\mathrm{T}}$.

Now that the multivariate constraints have been converted to univariate constraints in Eqn. (32), they can be efficiently evaluated through,

$$\begin{aligned} \mathrm{P}(a_i^{\mathrm{T}} \mathbb{X} > b_i) \quad &= \mathrm{P}(y_i > b_i) \\ &= \frac{1}{\sqrt{2\pi a_i^{\mathrm{T}} \mathbb{M} a_i}} \int_{b_i}^{\infty} \exp\left(-\frac{(y_i - a_i^{\mathrm{T}} \bar{\mathbb{X}})^2}{2 a_i^{\mathrm{T}} \mathbb{M} a_i}\right) dy_i \\ &= \frac{1}{\sqrt{2\pi}} \int_{\frac{b_i - a_i^{\mathrm{T}} \bar{\mathbb{X}}}{\sqrt{a_i^{\mathrm{T}} \mathbb{M} a_i}}}^{\infty} \exp\left(-\frac{z^2}{2}\right) dz \\ &= 1 - \mathrm{normcdf}\left(\frac{b_i - a_i^{\mathrm{T}} \bar{\mathbb{X}}}{\sqrt{a_i^{\mathrm{T}} \mathbb{M} a_i}}\right). \end{aligned} \tag{34}$$

The function $\mathrm{normcdf}(\cdot)$ is the Gaussian cumulative distribution function,

$$\mathrm{normcdf}(x) = \frac{1}{\sqrt{2\pi}} \int_{-\infty}^{x} \exp\left(-\frac{z^2}{2}\right) dz. \tag{35}$$

Although the $\mathrm{normcdf}(\cdot)$ function does not have an analytic solution, it can be efficiently evaluated using a series approximation or a pre-computed lookup table.

B. Gradient and Hessian

The gradient and Hessian of the probability constraints can also be computed analytically to aid in the solution of the optimization problem. First, the gradient will be computed for one probability constraint and then generalized for all. From the chain rule,

$$\nabla_{x^d} \mathrm{P}\left(a_i^{\mathrm{T}} \mathbb{X} > b_i \right) = \frac{\partial \mathrm{P}\left(a_i^{\mathrm{T}} \mathbb{X} > b_i \right)}{\partial \bar{\mathbb{X}}} \nabla_{x^d} \bar{\mathbb{X}}. \tag{36}$$

From Eqn. (34) and the Leibniz integral rule,

$$\begin{aligned} \frac{\partial \mathrm{P}\left(a_i^{\mathrm{T}} \mathbb{X} > b_i \right)}{\partial \bar{\mathbb{X}}} \quad &= \quad -\frac{\partial \mathrm{normcdf}\left(\frac{b_i - a_i^{\mathrm{T}} \bar{\mathbb{X}}}{\sqrt{a_i^{\mathrm{T}} \mathbb{M} a_i}}\right)}{\partial \bar{\mathbb{X}}} \\ &= \quad \frac{1}{\sqrt{2\pi a_i^{\mathrm{T}} \mathbb{M} a_i}} \exp\left(-\frac{(b_i - a_i^{\mathrm{T}} \bar{\mathbb{X}})^2}{2 a_i^{\mathrm{T}} \mathbb{M} a_i}\right) a_i^{\mathrm{T}}. \end{aligned} \tag{37}$$

The gradient of $\bar{\mathbb{X}}$ with respect to x^d is $\nabla_{x^d} \bar{\mathbb{X}} = T^{xxd}$. Combining the individual gradients into vector notation yields,

$$g_c = {T^{xxd}}^{\mathrm{T}} A_c \phi \tag{38}$$

where ϕ is defined as a vector with elements,

$$\phi_i = \frac{1}{\sqrt{2\pi a_i^{\mathrm{T}} \mathbb{M} a_i}} \exp\left(-\frac{(b_i - a_i^{\mathrm{T}} \bar{\mathbb{X}})^2}{2 a_i^{\mathrm{T}} \mathbb{M} a_i}\right), \tag{39}$$

and $A_c = [a_1, \ldots, a_{N_{F_X}}]$. The Hessian of the probability constraints is determined through a similar process. The Hessian of one individual constraint is,

$$\nabla^2_{x_d} \mathrm{P}\left(a_i^{\mathrm{T}} \mathbb{X} > b_i\right) = d_i {T^{xx_d}}^{\mathrm{T}} a_i a_i^{\mathrm{T}} T^{xx_d} \tag{40}$$

where $d_i = \frac{b_i - a_i^{\mathrm{T}} \bar{\mathbb{X}}}{\sqrt{2\pi} a_i^{\mathrm{T}} \mathbb{M} a_i} \exp\left(-\frac{(b_i - a_i^{\mathrm{T}} \bar{\mathbb{X}})^2}{2 a_i^{\mathrm{T}} \mathbb{M} a_i}\right)$. The Hessian of all the probability constraints in matrix form is then,

$$H_c = {T^{xxd}}^{\mathrm{T}} A_c \mathrm{diag}\left(d\right) A_c^{\mathrm{T}} T^{xxd}. \tag{41}$$

VI. New Optimization Program

Now that the distribution of the closed-loop system state and the efficient evaluation of a single chance constraint have been developed, the satisfaction of the overall probability constraint, $\mathrm{P}(\mathbb{X} \notin F_X) \leq \delta$, needs to be addressed. Previously, two methods have been proposed for this: fixed risk [15] and risk allocation [10], [1].

The fixed risk method assigns a pre-defined allowed probability of violation for each univariate constraint, $\mathrm{P}(a_i^{\mathrm{T}} \mathbb{X} > b_i)$, such as δ / N_{F_X}. This pre-defined probability is chosen to ensure that the total probability of violation is below the threshold δ. Since the allowed risk for each constraint is known beforehand, the chance constraints can be simplified by modifying the system's feasible region at each time-step.

The risk allocation method includes the allowed probability of violation for each univariate constraint, $\mathrm{P}(a_i^{\mathrm{T}} \mathbb{X} > b_i)$, as an optimization variable, ϵ_i. In order to ensure that the total probability of violation is below the threshold, the optimization variables are restricted by $\sum \epsilon_i \leq \delta$. For this work, this risk allocation method was ultimately chosen as it provided increased system performance with only slight increases in computational complexity.

Using this risk allocation technique [10] along with the distribution of the closed-loop state in Section IV and the evaluation of the chance constraints in Section V, the Program P2.1 can now be transformed into the optimization Program P6.1.

$$\begin{aligned}
\text{minimize} \quad & f(\bar{\mathbb{X}}, \bar{\mathbb{U}}) \\
\text{subject to} \quad & \\
& \bar{\mathbb{X}} = T^{xx} \bar{x}_0 + T^{xxd} \mathbb{X}^d \\
& \bar{\mathbb{U}} = T^{ux_0} \bar{x}_0 + T^{uxd} \mathbb{X}^d \\
& \bar{\mathbb{U}} \in F_U \\
& z_i = b_i - a_i^{\mathrm{T}} \bar{\mathbb{X}}, \ \forall i \\
& \sigma_i = a_i^{\mathrm{T}} \mathbb{M} a_i, \ \forall i \\
& 1 - \mathrm{normcdf}\left(\tfrac{z_i}{\sqrt{\sigma_i}}\right) \leq \epsilon_i, \ \forall i \\
& \textstyle\sum_{i=1}^{N_{F_X}} \epsilon_i \leq \delta
\end{aligned} \tag{P6.1}$$

The convexity of program P6.1 can be shown by using the results established by Prékopa for probabilistically constrained stochastic optimization programs with log-concave measures [16] which is similarly used in [1] for the open-loop control problem.

Given the convexity of problem P6.1 there are many well known algorithms which can be used to solve for the globally optimal solution. For this work, a custom log barrier method with a Newton step was used. Also, the $\mathrm{normcdf}$ function was evaluated using a pre-computed lookup table.

VII. Examples

A. Open-Loop Unstable

Since the formulation incorporates a feedback controller instead of only designing the open-loop control inputs, unstable systems can be handled as long as a linear quadratic feedback law can stabilize the system.

For this example, the unstable system dynamics are,

$$A = \begin{bmatrix} 2.72 & 0 \\ 0.17 & 1 \end{bmatrix} \quad B = \begin{bmatrix} 0.17 \\ 7.2e-3 \end{bmatrix} \quad C = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}$$

and the noise parameters are, $\Sigma_w = 0.0001I$ and $\Sigma_v = 0.0001I$. The time horizon is $N = 20$ and the initial condition is assumed to be distributed according to $x_0 \sim \mathcal{N}\left(\begin{bmatrix} 0 \\ 0 \end{bmatrix}, 0.0001I\right)$. The feasible region, F_X, is defined by the following constraints,

$$\begin{bmatrix} 1 & 0 \end{bmatrix} x_k \leq 1.05 \quad \begin{bmatrix} -1 & 1 \end{bmatrix} x_k \leq 0.3$$

for all $k \in [1, \ldots, N]$. The allowed probability constraint violation is $\delta = 0.01$. The objective function for this problem is quadratic in the final state as well as the control inputs,

$$f(\bar{\mathbb{X}}, \bar{\mathbb{U}}) = \left(x_N - x_{ref}\right)^{\mathrm{T}} Q_{obj} \left(x_N - x_{ref}\right) + \bar{\mathbb{U}}^{\mathrm{T}} R_{obj} \bar{\mathbb{U}} \tag{42}$$

with $Q_{obj} = I$, $R_{obj} = 0.001I$ and $x_{ref} = \begin{bmatrix} 1 & 1 \end{bmatrix}^{\mathrm{T}}$.

The solution for this example is shown in Figure 1 in which the line is the trajectory of the system and the ellipses are the 99.7% confidence ellipsoid at each time-step. The optimal objective function for this problem is 0.045 and was solved in 15 milliseconds. Figure 2 displays the constraint violation probability at each time-step for the slanted line. At the start of the trajectory, the system has a very low chance of violating the constraints, but near the end the probability of violation increases dramatically. This non-uniform distribution of constraint violation probability is a result of using the risk allocation algorithm, which in this case has chosen to assign most of the allowable constraint violation near the end of the trajectory. By using this risk allocation algorithm, the performance of the system was increased as compared with using a pre-defined risk profile.

B. Non-Convex Environments

If the feasible region, F_X, is a non-convex region then there are several ways to transform it into a series of convex problems [14], [17]. The approach taken in the following examples was to decompose the feasible region into convex

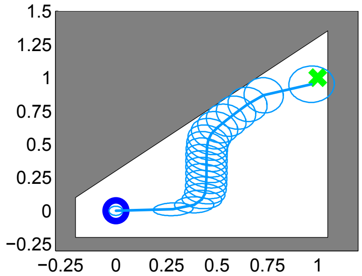


Fig. 1. The solution for the unstable dynamics example. The blue circle is the starting position of the state and the green 'x' is the goal location. The white area is the feasible region F_X. The ellipses are the 99.7% confidence regions of the state at each time-step.

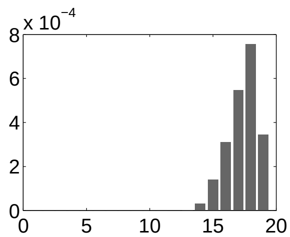


Fig. 2. The probability of violating the slanted line constraint for the open-loop unstable example. The x-axis is the time-step and the y-axis is the probability of violating the constraint.

tunnels which are then planned through. The system has double integrator dynamics with a 2D position,

$$A = \begin{bmatrix} 1 & 0 & \Delta t & 0 \\ 0 & 1 & 0 & \Delta t \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}, \quad B = \begin{bmatrix} 0.5\Delta t^2 & 0 \\ 0 & 0.5\Delta t^2 \\ \Delta t & 0 \\ 0 & \Delta t \end{bmatrix}. \qquad C = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \end{bmatrix} \tag{43}$$

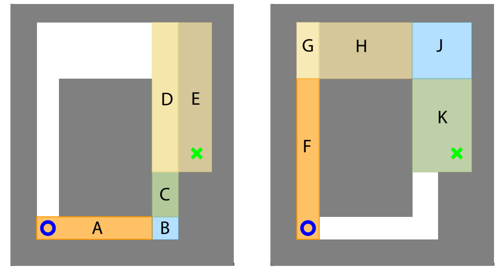
where $\Delta t = 0.1$ seconds and a time-horizon of $N = 20$. The noise parameters are,

$$\begin{aligned} \Sigma_w &= \mathrm{diag}(0.0003, 0.0005, 0.0003, 0.0005) \\ \Sigma_v &= \mathrm{diag}(0.001, 0.002) \end{aligned} \tag{44}$$

The objective function is the same as the previous example in Eqn. (42) with $x_{ref} = \begin{bmatrix} 2 & 1 \end{bmatrix}^{\mathrm{T}}$. The constraints that exist for all the different paths from the start to the goal are,

$$\begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ -1 & 0 & 0 & 0 \\ 0 & -1 & 0 & 0 \end{bmatrix} x_k \le \begin{bmatrix} 2.2 \\ 2.75 \\ 0.15 \\ 0.15 \end{bmatrix}.$$

The feasible region can be decomposed into two different tunnels corresponding to going to the top or bottom of the first obstacle. Figure 3 shows the decomposition of the non-convex region for the top and bottom paths.



(a) (b)

Fig. 3. The decomposition of the non-convex space into tunnels corresponding to the bottom path and top path in (a) and (b), respectively.

1) Bottom Region: The probability constraints for the bottom region are,

$$\begin{aligned} \begin{bmatrix} 0 & 1 & 0 & 0 \end{bmatrix} x_k &\le 0.15 && \forall k \in [1, k_1] \\ \begin{bmatrix} -1 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 \end{bmatrix} x_k &\le \begin{bmatrix} -1.4 \\ 1.75 \end{bmatrix} && \forall k \in [k_1 + 1, k_2] \\ \begin{bmatrix} 0 & -1 & 0 & 0 \end{bmatrix} x_k &\le -0.75 && \forall k \in [k_2 + 1, N] \end{aligned}$$

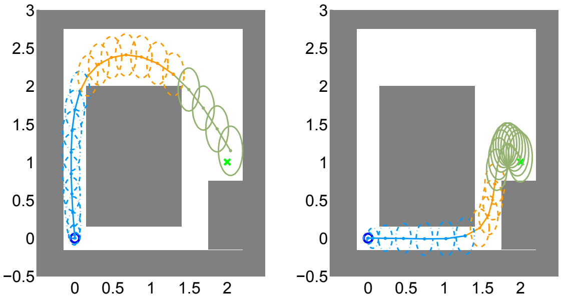
where k_1 and k_2 are the switching times between the three sets of constraints. The switching times corresponds when the system exits one region, defined by the constraints, and enters another. Corresponding to the regions in Figure 3(a), the first inequality constrains the system to be in regions A and B, the second inequality restricts the system to only be in regions B, C and D, and the final constraint allows the system to be in regions D and E.

2) Top Region: Similarly, the probability constraints for the top region are,

$$\begin{aligned} \begin{bmatrix} 1 & 0 & 0 & 0 \end{bmatrix} x_k &\le 0.15 && \forall k \in [1, k_1] \\ \begin{bmatrix} 0 & -1 & 0 & 0 \end{bmatrix} x_k &\le -2 && \forall k \in [k_1{+}1, k_2] \\ \begin{bmatrix} -1 & 0 & 0 & 0 \\ 0 & -1 & 0 & 0 \end{bmatrix} x_k &\le \begin{bmatrix} -1.4 \\ -0.75 \end{bmatrix} && \forall k \in [k_2{+}1, N] \end{aligned}$$

where again k_1 and k_2 are the switching times between the three sets of constraints. In reference to the regions in Figure 3(b), the first inequality constrains the system to be in regions F and G, the second inequality restricts the system to only be in regions G, H and J, and the final constraint allows to the system to be in regions J and K.

3) Results: For this example, the optimal path is always through the top region even though the bottom region is shorter. The bottom region is infeasible with respect to the chance constraints if the allowed constraint violation is $\delta < 0.19$ which is caused by the large uncertainty of the vertical position of the state. Figure 4(a) displays the optimal solution for $\delta = 0.1$ which travels through the top region and the optimal switching times are $k_1 = 9$ and $k_2 = 16$. The computed constraint violation from Monte Carlo simulations was 0.095, and the conservativeness is due to Boole's inequality. Notice how the optimal solution curves toward the infeasible region in the beginning instead of following the center of the corridor. This initial deviation improves the overall objective and is allowed because the rest of the path has a small probability of violating the constraints. Figure 4(b) shows a suboptimal solution through the bottom region for an allowed constraint violation of 0.19. The system follows the center of the corridor in the beginning and waits to reach the goal location at the end to reduce the probability of violating any constraints before the end of the planning horizon.



(a) (b)

Fig. 4. The results for planning through the non-convex region. The blue line with dash-dot ellipses, orange line with dash ellipses, and green line with solid ellipses indicate which three sets of constraints are active. (a) The optimal path for a constraint violation of $\delta = 0.1$ (b) Planning only through the bottom path with an allowed constraint violation of $\delta = 0.19$.

C. Collision Avoidance

For the final set of results, a dynamic non-convex example is chosen in which the vehicles have to avoid each other while attempting to reach their goal location. For each scenario, all vehicles have the same double integrator dynamics describe by Eqn. (43) with $\Delta t = 0.1$ seconds, a time-horizon of $N = 20$ and noise parameters given in Eqn. (44). In each scenario, the vehicles where required to keep a minimum separation distance of 0.25 meters, which by definition is a non-convex feasible region. Consequently, the same approach will be employed as in the previous section to simplify the problem into convex subproblems.

The objective function for each of the collision avoidance scenarios is,

$$f(\bar{\mathbb{X}}, \bar{\mathbb{U}}) = \sum_{i=1}^{N_V} \left(\bar{x}_N^i - x_{ref}^i\right)^{\mathrm{T}} Q_{obj} \left(\bar{x}_N^i - x_{ref}^i\right) + \bar{\mathbb{U}}^{\mathrm{T}} R_{obj} \bar{\mathbb{U}}$$

with $Q_{obj} = \begin{bmatrix} I & 0 \\ 0 & 0 \end{bmatrix}$, $R_{obj} = 0.001I$, $\bar{x}_k^i$ is the state mean of vehicle i at the k-th time-step, x_{ref}^i is the reference state of vehicle i, and N_V is the number of vehicles.

1) 2-vehicle: Due to the symmetric nature of the example, the problem can be simplified while still retaining the optimal solution. To this end, the vehicle which starts on the left will be constrained to pass under the other vehicle. The decomposed constraints are,

$$\begin{aligned}
(a)&\left\{\begin{bmatrix} 1 & 0 & 0 & 0 \end{bmatrix} (x_k^1 - x_k^2) \leq -0.25 \quad \forall k \in [1, k_1]\right. \\
(b)&\left\{\begin{bmatrix} 0 & 1 & 0 & 0 \end{bmatrix} (x_k^1 - x_k^2) \leq -0.25 \quad \forall k \in [k_1{+}1, k_2]\right. \\
(c)&\left\{\begin{bmatrix} 1 & 0 & 0 & 0 \end{bmatrix} (x_k^2 - x_k^1) \leq -0.25 \quad \forall k \in [k_2{+}1, N]\right.
\end{aligned}$$

where x_k^i is the state of vehicle i at the k-th time-step with $i \in [1, 2]$. The constraints require the vehicle's position in the x-direction, then y-direction and finally x-direction to be greater than the required separation distance. Using this decomposition the optimal solution will be achieved as long as the optimal switching times k_1 and k_2 are determined, which they can be through a simple brute force search.

The starting and goal locations for each of the vehicles are, $x_0^1 = \begin{bmatrix} 0 & 0 & 0 & 0 \end{bmatrix}^{\mathrm{T}}$, $x_0^2 = \begin{bmatrix} 3 & 0 & 0 & 0 \end{bmatrix}^{\mathrm{T}}$, $x_{ref}^1 = x_0^2$ and $x_{ref}^2 = x_0^1$. Figure 5 shows the optimal solution with $k_1 = 12$ and $k_2 = 13$. The smallest separation between the two vehicles is 0.4 meters when they pass over each other in the middle.

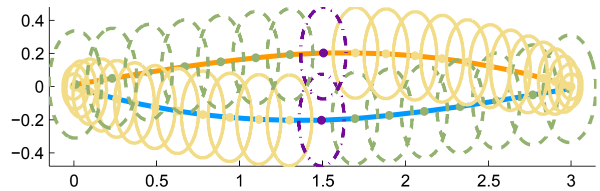


Fig. 5. Two vehicle collision avoidance scenario. The orange and blue lines represent the path of the two vehicles. The yellow solid, purple dash-dot, and green dash-dash ellipses indicate which set of constraints are active, (a), (b), (c), respectively.

2) 3-vehicle: The starting and goal locations for each of the vehicles are $x_0^1 = \begin{bmatrix} -0.87 & 0.5 & 0 & 0 \end{bmatrix}^{\mathrm{T}}$, $x_0^2 = \begin{bmatrix} 0.87 & 0.5 & 0 & 0 \end{bmatrix}^{\mathrm{T}}$ $x_0^3 = \begin{bmatrix} 2 & 0 & 0 & 0 \end{bmatrix}^{\mathrm{T}}$, $x_{ref}^1 = \begin{bmatrix} 0.5 & 1.87 & 0 & 0 \end{bmatrix}^{\mathrm{T}}$, $x_{ref}^2 = \begin{bmatrix} -0.5 & 1.87 & 0 & 0 \end{bmatrix}$ and $x_{ref}^3 = \begin{bmatrix} 0 & 0 & 0 & 0 \end{bmatrix}^{\mathrm{T}}$. Due to the large number of feasible trajectories through the environment, the allowed feasible region was restricted to reduce the computation time. Consequently, only a locally optimal solution can be guaranteed but it is the optimal solution within the restricted feasible region. The decomposed regions are defined by,

$$\begin{aligned}
(a)&\left\{\begin{array}{l}
\begin{bmatrix} 1 & 0 & 0 & 0 \end{bmatrix} (x_k^1 - x_k^2) \leq -0.25 \quad \forall k \in [1, k_1] \\
\begin{bmatrix} 0 & 1 & 0 & 0 \end{bmatrix} (x_k^1 - x_k^3) \leq -0.25 \\
\begin{bmatrix} 0 & 1 & 0 & 0 \end{bmatrix} (x_k^2 - x_k^3) \leq -0.25
\end{array}\right. \\
(b)&\left\{\begin{array}{l}
\begin{bmatrix} 0 & 1 & 0 & 0 \end{bmatrix} (x_k^1 - x_k^2) \leq -0.25 \quad \forall k \in [k_1{+}1, k_2] \\
\begin{bmatrix} 1 & 0 & 0 & 0 \end{bmatrix} (x_k^3 - x_k^1) \leq -0.25 \\
\begin{bmatrix} 1 & 0 & 0 & 0 \end{bmatrix} (x_k^3 - x_k^2) \leq -0.25
\end{array}\right. \\
(c)&\left\{\begin{array}{l}
\begin{bmatrix} 0 & 1 & 0 & 0 \end{bmatrix} (x_k^3 - x_k^1) \leq -0.25 \quad \forall k \in [k_2{+}1, N] \\
\begin{bmatrix} 0 & 1 & 0 & 0 \end{bmatrix} (x_k^3 - x_k^2) \leq -0.25 \\
\begin{bmatrix} 1 & 0 & 0 & 0 \end{bmatrix} (x_k^2 - x_k^1) \leq -0.25
\end{array}\right.
\end{aligned}$$

where again k_1 and k_2 are the times when the vehicle exits and enters the different regions. In words the constraints are as follows: (a) requires vehicle 3 to be above vehicles 1, 2 and vehicle 1 to be to the left of vehicle 2, (b) requires vehicle 3 to be to the left of vehicles 1, 2 and vehicle 2 is above vehicle 1, (c) requires vehicle 3 to be below vehicles 1, 2 and vehicle 2 is to the left of vehicle 1.

Figure 6 shows a series of images of the vehicle locations for the expected trajectories for a problem with an allowed constraint violation of $\delta = 0.1$. For each of the subplots in the figure, the system is in regions (a),(b),(b),(c), respectively, of the decomposed environment. Statistics for the two and three vehicle collision avoidance examples are shown in Table I. As the allowed constraint violation increases, the performance of the system increases as expected. The table also

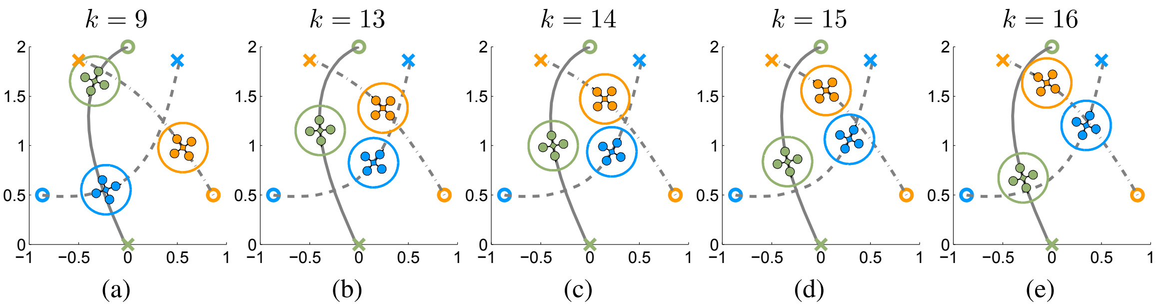


Fig. 6. A three vehicle collision avoidance scenario with an allowed constraint violation of $\delta = 0.1$. In each plot, the vehicles are attempting to reach their goal location marked by the 'x'.

shows the simulated constraint violation for each scenario, and due to the approximation from Boole's inequality the solutions are slightly conservative.

TABLE I

A COMPARISON OF THE SOLUTION STATISTICS AND COMPUTATION TIME FOR THE TWO AND THREE VEHICLE COLLISION AVOIDANCE EXAMPLES.

# Vehicles	δ	δ_{sim}	Sol. Time (sec.)	f^*	k_1	k_2
2	0.001	9.2e-4	9.5	0.093	11	13
2	0.01	0.009	8.9	0.087	11	13
2	0.1	0.096	9.6	0.078	12	13
3	0.001	9e-4	29.3	0.1582	9	16
3	0.01	9.9e-3	30.1	0.1311	9	15
3	0.1	0.083	30.5	0.1061	10	15

VIII. CONCLUSIONS

The motion planning problem for a linear, Gaussian stochastic system with system state constraints was formulated as an optimal control problem. The near-optimal solution was obtained by solving a conservative convex optimization problem that restricted the probability of constraint violation for the closed-loop system to be below a pre-defined threshold. This threshold is a tuning parameter which trades-off the performance the conservativeness of the solution.

There are several interesting areas of future work that the authors wish to explore. This work could be used as the basis for developing heuristics to solve for locally optimal policies for more complex systems such as nonlinear dynamics and/or nonlinear sensors. Another area of future work would be to apply the algorithm online in a receding horizon fashion. While the constraints would only be guaranteed to hold for the finite horizon case, the performance of the system would be increased by using the *a posterior* distribution of the system state online. Lastly, the authors wish to apply these algorithms to actual systems performing tasks such as collision avoidance, multiagent perception, search and rescue, and environment discovery.

REFERENCES

[1] L. Blackmore and M. Ono, "Convex chance constrained predictive control without sampling," in *Proceedings of the AIAA Guidance, Navigation, and Control Conference*, Chicago, Illinois, August 2009.

[2] D. van Hessem, "Stochastic inequality constrained closed-loop model predictive control with application to chemical process operation," Ph.D. dissertation, Delft University of Technology, 2004.

[3] N. Melchior and R. Simmons, "Particle RRT for path planning with uncertainty," in *Proceedings IEEE International Conference on Robotics and Automation*, Roma, Italy, April 2007.

[4] R. Alterovitz, T. Simeon, and K. Goldberg, "The stochastic motion roadmap: A sampling framework for planning with markov motion uncertainty," in *Proceedings of Robotics: Science and Systems*, Atlanta, GA, USA, June 2007.

[5] R. Platt, R. Tedrake, L. Kaelbling, and T. Lozano-Perez, "Belief space planning assuming maximum likelihood observations," in *Proceedings of Robotics: Science and Systems*, Zaragoza, Spain, June 2010.

[6] V. A. Huynh and N. Roy, "icLQG: Combining local and global optimization for control in information space," in *Proceedings of the IEEE International Conference on Robotics and Automation*, Kobe, Japan, May 2009.

[7] J. Gonzalez and A. Stentz, "Using linear landmarks for path planning with uncertainty in outdoor environments," in *Proceedings IEEE/RSJ International Conference on Intelligent Robots and Systems*, 2009.

[8] R. Pepy and A. Lambert, "Safe path planning in an uncertain-configuration space using RRT," in *Proceedings IEEE/RSJ International Conference on Intelligent Robots and Systems*, 2006.

[9] L. Blackmore, "A probabilistic particle control approach to optimal, robust predictive control," in *Proceedings of the AIAA Guidance, Navigation, and Control Conference*, Keystone, Colorado, August 2006.

[10] A. Prékopa, "The use of discrete moment bounds in probabilistic constrained stochastic programming models," *Annals of Operations Research*, vol. 85, pp. 21–38, 1999.

[11] M. Kothare, V. Balakrishnan, and M. Morari, "Robust constrained model predictive control using linear matrix inequalities," *Automatica*, vol. 32, no. 10, pp. 1361–1379, 1996.

[12] D. Mayne, J. Rawlings, C. Rao, and P. Scokaert, "Constrained model predictive control: Stability and optimality," *Automatica*, vol. 36, no. 6, pp. 789–814, 2000.

[13] J. van den Berg, P. Abbeel, and K. Goldberg, "LQG-MP: Optimized path planning for robots with motion uncertainty and imperfect state information," in *Proceedings of Robotics: Science and Systems*, Zaragoza, Spain, June 2010.

[14] M. P. Vitus, S. L. Waslander, and C. J. Tomlin, "Locally optimal decomposition for autonomous obstacle avoidance with the tunnel-MILP algorithm," in *Proceedings of the 47th IEEE Conference on Decision and Control*, Cancun, Mexico, December 2008.

[15] A. Nemirovksi and A. Shapiro, "Convex approximations of chance constrained programs," *SIAM Journal of Optimization*, vol. 17, no. 4, pp. 969–996, 2006.

[16] A. Prékopa, *Stochastic Programming*. Kluwer Scientific, 1995.

[17] M. Ono, L. Blackmore, and B. C. Williams, "Chance constrained finite horizon optimal control with nonconvex constraints," in *Proceedings of the 2010 American Control Conference*, Baltimore, Maryland, June 2010.