Belief Space Planning For Linear, Gaussian Systems In Uncertain Environments

Michael P. Vitus * **Claire J. Tomlin** **

* *Department of Aeronautics and Astronautics, Stanford University, USA (e-mail: vitus@stanford.edu).*
** *Department of Electrical Engineering and Computer Sciences, University of California at Berkeley, USA (e-mail: tomlin@eecs.berkeley.edu)*

Abstract: This paper introduces a new formulation for solving the planning problem for linear, Gaussian stochastic systems through uncertain environments. Due to the imperfect knowledge of the system state caused by motion uncertainty, sensor noise and environment uncertainty, the system constraints cannot be guaranteed to be satisfied and consequently must be considered probabilistically. For a known environment, the constraints can be modeled as conjunctions, unions of convex expressions of univariate Gaussian random variables. However, for uncertain environments the constraints transform into the sums of products of random variables which do not have a closed-form analytical expression. In general, to calculate the probability of constraint violation a sampling method would have to be used to evaluate the multivariate integrals; this sampling strategy would be computationally expensive and possibly intractable for higher dimensional systems. Fortunately, even for low-dimensional systems the constraints are shown to be very well approximated by a univariate Gaussian distribution and can be evaluated efficiently. The violation probability of all the constraints is guaranteed to be below a threshold that trades off the performance of the system and the conservativeness of the solution. In contrast to similar methods, the proposed framework considers the specific estimator and controller used in the closed-loop system in order to fully characterize the *a priori* distribution of the closed-loop system state. Using this distribution, an optimization program is formulated to solve for a locally optimal solution for the closed-loop system. The performance of the algorithm is demonstrated through an example.

Keywords: Motion Planning, Chance Constraints, Robust Control, Probabilistic Risk Assessment.

1. INTRODUCTION

With the advances in technology of robotic systems, there has been a growing number of robots deployed to perform challenging tasks such as: search and rescue, reconnaissance, surveillance, and disaster response. In all of these tasks, there is a pressing need for algorithms to plan safe trajectories through complex environments. The general motion planning problem involves generating a trajectory that accomplishes a defined task while operating under specified system dynamics and constraints. In the planning process the system cannot be assumed to be deterministic, rather the inherit uncertainty of the system must be accounted for explicitly in order to maximize the success of the resulting plan.

The uncertainty in the planning problem arises from three different sources: (i) motion uncertainty, (ii) sensing noise and (iii) environment uncertainty. The presence of these uncertainties means that 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, 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.

The problem of motion planning with motion noise, sensing noise and/or environment uncertainty has been studied in the past. Some previous planners (Melchior and Simmons (2007), Alterovitz et al. (2007)) 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 (Huynh and Roy (2009), Pepy and Lambert (2006)) 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 system state which can lead to a violation of the constraints on the system.

Charnes and Cooper (1963) introduced the problem of *chance constrained programming* which allows for non-deterministic constraint parameters, while only guaranteeing constraint satisfaction up to a specified limit. A thor-

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

ough account of existing literature employing this problem formulation is given in Prékopa (1995). Recently, Calafiore and El Ghaoui (2007) investigated an inequality chance constrained linear program with uncertain constraint parameters. They showed that for a wide class of probability distributions the problem can be converted into a second order cone program. The problem considered in this paper is related to this work, with the added difficulty of incorporating probabilistic constraint variables.

A group of researchers used the chance constrained programming formulation to model the planning problem as an optimal control problem. In Blackmore (2006), they handle non-Gaussian belief distributions by approximating them 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 represent the true belief state increases. The work by Blackmore and Ono (2009) utilized Boole's inequality to approximate the chance constraints, typically resulting in a very small amount of over-conservativeness. They also used the idea of risk allocation introduced by Prékopa (1999) to distribute the risk of violating each chance constraint while still guaranteeing safety.

The work by van Hessem (2004) optimized over the feedback control laws and open-loop inputs while ensuring that the chance constraints on the overall system were satisfied. They used an ellipsoidal relaxation technique to convert the stochastic problem into a deterministic one but this leads to a conservative solution.

Incorporating environment uncertainty into the motion planning problem has also received some attention. Missiuro and Roy (2006) handled uncertain environments by modifying the sampler used in a probabilistic roadmap. However, while the algorithm accounts for the environment uncertainty, motion noise or sensing noise is not accounted for. Burns and Brock (2007) also proposed a sampling-based planner to incorporate environment sensing uncertainty into the planning process but do not directly handle motion noise.

This work extends previous chance constrained programming formulations to solve the motion planning problem in uncertain environments. A framework is proposed that formulates the motion planning problem for a linear, Gaussian system operating in an uncertain environment as an optimal control problem. The probabilistic constraints on the system state are shown to be distributed via the sums of products of random variables. In general the constraint expression does not have a closed-form analytical expression and requires the evaluation of multivariate integrals. One way to calculate the probability of constraint violation is through sampling but it is computationally expensive and becomes intractable as the system size increases. Fortunately, it is shown that even for low-dimensional systems the expression is very well approximated by a univariate Gaussian distribution due to the central limit theorem. Consequently this representation allows for the efficient computation of the probability of constraint violation for otherwise intractable problems. The formulation also considers the closed-loop uncertainty directly to solve for a locally optimal desired path for the feedback controller.

The paper proceeds as follows. Section 2 describes the probabilistic problem formulation. Then, the explicit estimator and controller used in the closed-loop system is formally presented in Section 3. In Section 4, the *a priori* uncertainty of the closed-loop system is derived which is used in evaluating the chance constraints presented in Section 5. The final optimization program is presented in Section 6, and an example is presented in Section 7 which characterizes the performance of the algorithm.

2. 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, $\mathbb{X} = \begin{bmatrix} x_1^{\mathrm{T}} & \dots & x_N^{\mathrm{T}} \end{bmatrix}^{\mathrm{T}}$ and $\mathbb{U} = \begin{bmatrix} u_0^{\mathrm{T}} & \dots & u_{N-1}^{\mathrm{T}} \end{bmatrix}^{\mathrm{T}}$. 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.

For simplicity, the feasible region F_X is assumed to be convex. Nonconvex regions can still be handled, however, either by (i) performing branch and bound on the set of conjunction and disjunction linear state constraints directly, or by (ii) decomposing the space into convex regions and using branch and bound to determine when to enter/exit each convex subregion. Given this assumption, the feasible region can be defined by a conjunction of N_{F_X} linear inequality constraints,

$$F_X \triangleq \bigcap_{i=1}^{N_{F_X}} \left\{ \mathbb{X} : a_i^{\mathrm{T}} \mathbb{X} \le b_i \right\} \tag{3}$$

where $a_i \in \mathbb{R}^{nN}$ and $b_i \in \mathbb{R}$. In this work, the environment is uncertain but the parameters of the probability distribution describing a_i and b_i are assumed to be known.

The general belief space planning problem is posed as the optimization program, shown as Program P2.1. The objective function, $f(\cdot)$, is assumed to be a convex function.

Motion Planning Problem

minimize $\mathbf{E}\left(f(\mathbb{X}, \mathbb{U})\right)$

subject to

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

The difficulty in solving the optimization program P2.1 is in evaluating the chance constraints: $\mathrm{P}(\mathbb{X} \notin F_X) \leq \delta$. The complexity arises from the need to characterize the probability distribution of the system state at each time-step and from evaluating the multivariate integrals over the uncertain environment to calculate the desired probability of failure.

Characterizing the closed-loop probability distribution is particularly 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 specified to allow characterization of the uncertainty of the closed-loop system state.

3. 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.

3.1 Kalman Filter

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,

$$\hat{x}_{k|k-1} = A\hat{x}_{k-1|k-1} + Bu_{k-1} \tag{4}$$

$$\hat{\Sigma}_{k|k-1} = A\hat{\Sigma}_{k-1|k-1}A^{\mathrm{T}} + \Sigma_w. \tag{5}$$

The measurement update is,

$$\hat{x}_{k|k} = \hat{x}_{k|k-1} + L_k\left(y_k - C\hat{x}_{k|k-1}\right) \tag{6}$$

$$\hat{\Sigma}_{k|k} = (I - L_kC)\,\hat{\Sigma}_{k|k-1} \tag{7}$$

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

$$L_k = \hat{\Sigma}_{k|k-1}C^{\mathrm{T}}\left(C\hat{\Sigma}_{k|k-1}C^{\mathrm{T}} + \Sigma_v\right)^{-1}. \tag{8}$$

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 motion planning process.

3.2 Linear Quadratic Gaussian Control

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

$$J = \underset{u_k, \forall k\in T_N}{\text{minimize}} \sum_{\tau=0}^{N}\left(x_\tau - x_\tau^d\right)^{\mathrm{T}}Q\left(x_\tau - x_\tau^d\right) + \sum_{\tau=0}^{N-1}u_\tau^{\mathrm{T}}Ru_\tau, \tag{9}$$

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,

$$u_k^* = K_kx_k + g_k. \tag{10}$$

The affine parameters, K_k and g_k, can be computed by solving Eqn. (9) through dynamic programming.

Given that there is process and measurement noise, the exact value of the state x_k is not known. 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. (10), i.e.,

$$u_k^* = K_k\hat{x}_{k|k} + g_k. \tag{11}$$

4. *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 van den Berg et al. (2010), 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 and estimated state evolves according to,

$$z_{k+1} = F_kz_k + \bar{B}g_k + G_ks_k \tag{12}$$

where

$$z_k = \begin{bmatrix} x_k \\ \hat{x}_{k|k}\end{bmatrix},\ s_k = \begin{bmatrix} w_k \\ v_{k+1}\end{bmatrix}\ \bar{B} = \begin{bmatrix} B \\ B\end{bmatrix},$$
$$F_k = \begin{bmatrix} A & BK_k \\ L_{k+1}CA & A + BK_k - L_{k+1}CA\end{bmatrix},\ G_k = \begin{bmatrix} I & 0 \\ L_{k+1}C & L_{k+1}\end{bmatrix} \tag{13}$$

with $s_k \sim \mathcal{N}(0, \Sigma_s)$ and $\Sigma_s = \mathrm{diag}(\Sigma_w, \Sigma_v)$ (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. (12) can now be determined,

$$\bar{z}_{k+1} = F_k\bar{z}_k + \bar{B}g_k \tag{14}$$

$$M_{k+1} = F_kM_kF_k^{\mathrm{T}} + G_k\Sigma_sG_k^{\mathrm{T}}, \tag{15}$$

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,

$$\begin{bmatrix} x_k \\ u_k\end{bmatrix} = \underbrace{\begin{bmatrix} I & 0 \\ 0 & K_k\end{bmatrix}}_{\Lambda_k}\begin{bmatrix} x_k \\ \hat{x}_{k|k}\end{bmatrix} + \begin{bmatrix} 0 \\ I\end{bmatrix}g_k. \tag{16}$$

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

$$\begin{bmatrix} x_k \\ u_k\end{bmatrix} \sim \mathcal{N}\left(\begin{bmatrix} T_k^{xx}\bar{x}_0 + T_k^{xg}\mathbf{g} \\ K_k(T_k^{xx}\bar{x}_0 + T_k^{xg}\mathbf{g}) + g_k\end{bmatrix}, \Lambda_kM_k\Lambda_k^{\mathrm{T}}\right) \tag{17}$$

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 $T_k^{xx} = \prod_{\tau=0}^{k-1}(A + BK_\tau)$ and the elements of the matrix T^{xg} are,

$$T_{ij}^{xg} = \begin{cases} 0 & \text{if } j > i \\ B & \text{if } j = i \\ (\prod_{\tau=j+1}^{i}(A + BK_\tau))B & \text{otherwise}\end{cases} \tag{18}$$

The constant input term in Eqn. (11) 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 $\mathbb{Q}$ can be written in terms of the desired trajectory, $\mathbb{Q} = \Phi\mathbb{X}^d$ where Φ is defined as $\forall i, j \in [0, N-1]$,

$$\Phi_{ij} = \begin{cases} 0 & \text{if } j < i \\ -Q^{\mathrm{T}} & \text{if } j = i \\ -(\prod_{\tau=i+1}^{j}(A + BK_\tau)^{\mathrm{T}})Q^{\mathrm{T}} & \text{otherwise.} \end{cases} \quad (19)$$

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

$$\bar{\mathbb{X}} = T^{xx}\bar{x}_0 + T^{xxd}\mathbb{X}^d \quad (20)$$

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

$$\bar{\mathbb{U}} = T^{ux_0}\bar{x}_0 + T^{uxd}\mathbb{X}^d \quad (21)$$

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.

5. CHANCE CONSTRAINTS

Using the probability distribution of the closed-loop system state formulated in the previous section, the chance constraints in Program P2.1 can be evaluated efficiently once the multivariate integrals are simplified.

5.1 Simplification

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 $\mathrm{P}\left(\bigcup_i E_i\right) \leq \sum_i \mathrm{P}\left(E_i\right)$. Consequently, from Eqn. (3) and Boole's inequality the probability of the state not being contained inside the feasible region is bounded by,

$$\begin{aligned} \mathrm{P}(\mathbb{X} \notin F_X) &= \mathrm{P}\left(\mathbb{X} \in \bigcup_{i=1}^{N_{F_X}} \left\{\mathbb{X} : a_i^{\mathrm{T}}\mathbb{X} > b_i\right\}\right) \\ &\leq \textstyle\sum_{i=1}^{N_{F_X}} \mathrm{P}(a_i^{\mathrm{T}}\mathbb{X} > b_i). \end{aligned} \quad (22)$$

By allowing environmental uncertainty, a_i and b_i are now random variables, adding complexity to previous chance constrained problem formulations.

5.2 Environment Uncertainty

By allowing a_i and b_i to be uncertain, the distribution of $\sum_{j=1}^{nN} a_{ij}\mathbb{X}_j - b_i$ is now a sum of multiple products of random variables. Fortunately, the true distribution can be accurately approximated to allow the efficient evaluation of the constraints.

Due to the central limit theorem, if the dimension of the state space is large enough then the sum $\sum_{j=1}^{nN} a_{ij}\mathbb{X}_j - b_i$ can be approximated well by a univariate Gaussian distribution, eliminating the need to evaluate multivariate integrals. Given this conversion, the probability of constraint violation can now be evaluated efficiently.

If the system state and the environment are independent, then the mean of the distribution is,

$$E[a_i^{\mathrm{T}}\mathbb{X} - b_i] = E[a_i]^{\mathrm{T}}E[\mathbb{X}] - E[b_i] = \bar{a}_i^{\mathrm{T}}\bar{\mathbb{X}} - \bar{b_i} \quad (23)$$

and the covariance is,

$$\begin{aligned} \mathrm{var}(a_i^{\mathrm{T}}\mathbb{X} - b_i) &= \textstyle\sum_{j=1}^{nN} \mathrm{var}(a_{ij}\mathbb{X}_j) + \mathrm{var}(b_i) \\ &+ \textstyle\sum_{j,k}^{nN} \mathrm{cov}(a_{ij}\mathbb{X}_j, a_{ik}\mathbb{X}_k) \\ &+ \textstyle\sum_{j=1}^{nN} \mathrm{cov}(a_{ij}\mathbb{X}_j, b_i). \end{aligned} \quad (24)$$

To simplify later notation, let $\sigma_i^2 = \mathrm{var}(a_i^{\mathrm{T}}\mathbb{X} - b_i)$. One thing to note, is that the covariance might no longer be able to be computed *a priori* due to a dependence on the mean of the system state.

5.3 Univariate Gaussian Constraints

Using the conversion from the previous section, the constraints can now be evaluated using the distribution of the system state. From Eqns. (17) and (20) the closed-loop state is a Gaussian random variable defined by,

$$\mathbb{X} \sim \mathcal{N}\left(\bar{\mathbb{X}}, \mathbb{M}\right) \quad (25)$$

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

Since the multivariate constraints have been converted to univariate constraints in Eqn. (22) and can be approximated well as a Gaussian with mean and variance in Eqns. (23) and (24) respectively, they can be efficiently evaluated through,

$$\begin{aligned} \mathrm{P}(a_i^{\mathrm{T}}\mathbb{X} - b_i > 0) &= \frac{1}{\sqrt{2\pi}} \int_{\frac{\bar{b}_i - \bar{a}_i^{\mathrm{T}}\bar{\mathbb{X}}}{\sigma_i}}^{\infty} \exp(-\frac{z^2}{2})dz \\ &= 1 - \mathrm{normcdf}(\frac{\bar{b}_i - \bar{a}_i^{\mathrm{T}}\bar{\mathbb{X}}}{\sigma_i}) \end{aligned} \quad (26)$$

where $\mathrm{normcdf}(x) = \dfrac{1}{\sqrt{2\pi}} \int_{-\infty}^{x} \exp(-\dfrac{z^2}{2})dz$ is the Gaussian cumulative distribution function. Although this function does not have an analytic solution, it can be efficiently evaluated using a series approximation or a lookup table.

5.4 Gradient

The gradient of the probability constraints can also be computed analytically to aid in the solution of the optimization problem. The gradient can be computed through the chain rule,

$$\nabla_{x^d}\mathrm{P}\left(a_i^{\mathrm{T}}\mathbb{X} - b_i > 0\right) = \frac{\partial \mathrm{P}\left(a_i^{\mathrm{T}}\mathbb{X} - b_i > 0\right)}{\partial \bar{\mathbb{X}}} \nabla_{x^d}\bar{\mathbb{X}}. \quad (27)$$

From Eqn. (26) and the Leibniz integral rule,

$$\begin{aligned} &\frac{\partial \mathrm{P}\left(a_i^{\mathrm{T}}\mathbb{X} - b_i > 0\right)}{\partial \bar{\mathbb{X}}} = \\ &\frac{1}{\sqrt{2\pi}}\exp(-\frac{(b_i - a_i^{\mathrm{T}}\bar{\mathbb{X}})^2}{2\sigma_i^2}) \left[\frac{\bar{a}_i^{\mathrm{T}}}{\sigma_i} + \frac{\bar{b}_i - \bar{a}_i^{\mathrm{T}}\bar{\mathbb{X}}}{\sigma_i^2}\frac{\partial \sigma_i}{\partial \bar{\mathbb{X}}}\right]. \end{aligned} \quad (28)$$

Finally, the gradient of $\bar{\mathbb{X}}$ with respect to x^d is $\nabla_{x^d}\bar{\mathbb{X}} = T^{xxd}$ which results in,

$$\begin{aligned} &\nabla_{x^d}\mathrm{P}\left(a_i^{\mathrm{T}}\mathbb{X} - b_i > 0\right) = \\ &\frac{1}{\sqrt{2\pi}}\exp(-\frac{(b_i - a_i^{\mathrm{T}}\bar{\mathbb{X}})^2}{2\sigma_i^2})T^{xxd\mathrm{T}}\left[\frac{\bar{a}_i^{\mathrm{T}}}{\sigma_i} + \frac{\bar{b}_i - \bar{a}_i^{\mathrm{T}}\bar{\mathbb{X}}}{\sigma_i^2}\frac{\partial \sigma_i}{\partial \bar{\mathbb{X}}}\right]^{\mathrm{T}}. \end{aligned} \quad (29)$$

6. 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.

For this work, the risk allocation method (Blackmore and Ono (2009), Prékopa (1999)) was chosen as it provided increased system performance with only a slight increase in computational complexity. This 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 $\sum \epsilon_i \leq \delta$.

Using this risk allocation technique along with the distribution of the closed-loop state in Section 4 and the evaluation of the chance constraints in Section 5, the Program P2.1 can now be transformed into the optimization Program P6.1. The variable that is being optimized is the desired trajectory, $\mathbb{X}^d$, for the controller to follow.

$$\begin{aligned}
\underset{\mathbb{X}^d}{\text{minimize}} \quad & f(\bar{\mathbb{X}}, \bar{\mathbb{U}}) \\
\text{subject to} \quad & \\
& \bar{\mathbb{X}} = T^{xx}\bar{x}_0 + T^{xxd}\mathbb{X}^d \\
& \bar{\mathbb{U}} = T^{ux_0}\bar{x}_0 + T^{uxd}\mathbb{X}^d \\
& \bar{\mathbb{U}} \in F_U \\
& z_i = \bar{b}_i - \bar{a}_i^{\mathrm{T}}\bar{\mathbb{X}}, \ \forall i \\
& \sigma_i^2 = \mathrm{var}(a_i^{\mathrm{T}}\mathbb{X} - b_i), \ \forall i \\
& 1 - \mathrm{normcdf}(\tfrac{z_i}{\sigma_i}) \leq \epsilon_i, \ \forall i \\
& \textstyle\sum_{i=1}^{N_{F_X}} \epsilon_i \leq \delta
\end{aligned} \qquad \text{(P6.1)}$$

Lemma 1. The optimization program P6.1 is *not necessarily* convex if the objective function, $f(\cdot)$, is a convex function and $\delta \leq 0.5$.

Proof. To show the convexity of program P6.1, the objective function and all the constraints need to be convex. The objective function, $f(\cdot)$, is assumed to be a convex function with respect to $\bar{\mathbb{X}}$ and $\bar{\mathbb{U}}$. The first two constraints are convex because they are affine functions. The input constraint $\bar{\mathbb{U}} \in F_U$ is convex since the feasible region, F_U is assumed to be a convex region. The convexity of the program thus depends on the convexity of the chance constraints

In order for the chance constraints to be convex, the function $\mathrm{normcdf}(\cdot)$ must be concave. It can easily be shown that the function $\mathrm{normcdf}(y)$ is a concave, nondecreasing function on the domain $y \in [0, \infty)$. Thus in order for $\mathrm{normcdf}(\frac{z_i}{\sigma_i})$ to be concave, the argument must be both positive and concave (from convex composition rules). The positiveness is easily guaranteed by requiring $\delta \leq 0.5$. Thus the remaining requirement is for $\frac{z_i}{\sigma_i}$ to be concave.
The expression $\bar{b}_i - \bar{a}_i^{\mathrm{T}}\bar{\mathbb{X}}$ will preserve the convexity or concavity of $\dfrac{1}{\sigma_i}$. Therefore, the expression $\dfrac{1}{\sigma_i}$ needs to be shown to be concave. Unfortunately, the denominator could potentially be the composition of a concave, nondecreasing function (square root) with a convex expression (quadratic) of the state mean which is not guaranteed to be convex or concave on the entire domain. As a result, the program cannot be guaranteed to be convex. ∎

7. CASE STUDY: MOTION PLANNING IN A 2-DIMENSIONAL ENVIRONMENT

Now the optimization program P6.1 will be demonstrated through an example of a motion planning problem in a 2D Gaussian environment.

7.1 Environment Description

The environment is assumed to be defined by a set of half plane constraints defined by a series of end points. It is assumed that the end points for the i-th constraint e_1^i and e_2^i are independent, normally distributed,

$$e_j^i \sim \mathcal{N}\left(\begin{bmatrix} \bar{e}_{xj}^i \\ \bar{e}_{yj}^i \end{bmatrix}, \begin{bmatrix} \sigma_{e_{xj}^i}^2 & 0 \\ 0 & \sigma_{e_{yj}^i}^2 \end{bmatrix} \right), \forall i. \tag{30}$$

In an abuse of notation, let $p_k = [x_k \ y_k]^{\mathrm{T}}$ be the 2D position of the vehicle at time-step k with mean $\bar{p}_k = [\bar{x}_k \ \bar{y}_k]^{\mathrm{T}}$ and covariance $\mathrm{cov}(p_k) = \begin{bmatrix} \sigma_{x_k}^2 & \sigma_{x_k y_k}^2 \\ \sigma_{x_k y_k}^2 & \sigma_{y_k}^2 \end{bmatrix}$. The expression for the feasible region of the system state is,

$$F_X = \left\{ x : a_i^{\mathrm{T}} p_k \leq b_i, \forall (i,k) \in \mathcal{C} \right\} \tag{31}$$

where $\mathcal{C}$ is the set of constraints on the system. The vector a_i is defined as the outward pointing normal (and without loss of generality is assumed to be a 90^o counterclockwise rotation of the vector between the end points,

$$a_i = \begin{bmatrix} 0 & -1 \\ 1 & 0 \end{bmatrix} \begin{bmatrix} e_{x2}^i - e_{x1}^i \\ e_{y2}^i - e_{y1}^i \end{bmatrix} = \begin{bmatrix} e_{y1}^i - e_{y2}^i \\ e_{x2}^i - e_{x1}^i \end{bmatrix}. \tag{32}$$

The term b_i can be calculated using the normal vector, a_i, and one of the end points,

$$b_i = a_i^{\mathrm{T}} \left[e_{x2}^i \ e_{y2}^i \right]^{\mathrm{T}} = e_{x2}^i e_{y1}^i - e_{x1}^i e_{y2}^i. \tag{33}$$

The variance of the univariate chance constraint is,

$$\mathrm{var}(a_i^{\mathrm{T}} p_k - b_i) = \alpha_5 \bar{x}_k^2 + \alpha_6 \bar{x}_k + \alpha_7 \bar{y}_k^2 + \alpha_8 \bar{y}_k + \alpha_0, \tag{34}$$

and can be calculated through standard probability theory. The parameters α_i can be computed *a priori* but the variance depends upon the system state and consequently the control inputs, rendering the Program P6.1 nonconvex.

7.2 Validation Of The Gaussian Constraint Representation

A Monte Carlo simulation of the Gaussian approximation was performed in which the constraint and system parameters were generated in a way that the probability of constraint violation is 0.1. Figure 1 shows the resulting histogram of the value of the constraint from Monte Carlo simulation and the analytical Gaussian approximation. The solid line is the Monte Carlo simulation and the dotted line is the analytical Gaussian approximation. As the figure shows, the Gaussian approximation matches the Monte Carlo evaluation of the constraint. Another Monte Carlo

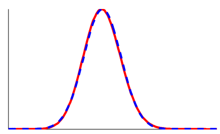

Fig. 1. Randomly generated constraint (solid) and the Gaussian approximation (dotted).

simulation of the Gaussian approximation was performed in which the constraint and system parameters were generated in a way that the probability of constraint violation is 0.1. The average Gaussian approximation error was 1.64%. The reason the Gaussian approximation overestimates the violation probability is because the tails of the distribution are difficult to capture through random sampling.

7.3 Example

The proposed motion planning method was used to plan a path through an environment that contains nonconvex regions which can be transformed into a series of convex problems Vitus et al. (2008), Ono et al. (2010). The approach taken in the following example was to decompose the feasible region into convex tunnels which are then planned through. The system has double integrator dynamics,

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

where $\Delta t = 0.1$ seconds and a time-horizon of $N = 20$. The noise parameters are $\Sigma_w = \text{diag}(0.001, 0.001, 0.001, 0.001)$, $\Sigma_v = \text{diag}(0.002, 0.002)$. The objective function for this problem is quadratic in the final state and the control inputs $f(\bar{\mathbb{X}}, \bar{\mathbb{U}}) = (\bar{x}_N - x_{ref})^{\mathrm{T}} Q_{obj} (\bar{x}_N - x_{ref}) + \bar{\mathbb{U}}^{\mathrm{T}} R_{obj} \bar{\mathbb{U}}$, with $Q_{obj} = 50I$, $R_{obj} = 0.001I$ and $x_{ref} = [\,2\ 1\ 0\ 0\,]^{\mathrm{T}}$.

The allowed constraint violation is $\delta = 0.005$ and the solution is shown in Figure 2. The solution which accounts for the uncertainty of the environment is shown as the blue, solid line and has a planning horizon of $N = 55$ with simulated constraint violation of 0.0038. The green, dotted line plans through the mean environment with a horizon of $N = 35$ and has a simulated constraint violation of 0.0084. When accounting for the uncertainty of the environment, the system cannot take the direct path as the mean environment solution can because the walls in that corridor have too much uncertainty resulting in the violation of the probability constraint.

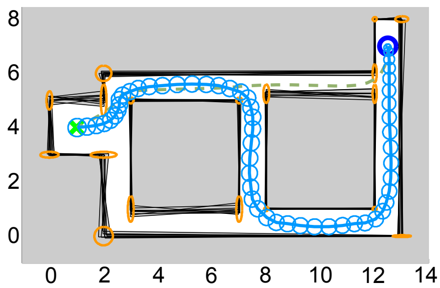

Fig. 2. The white area is the feasible region of the system state and the orange ellipses are the uncertainty of the environment. The blue, solid line is the solution when accounting for the uncertainty of the environment and the green, dotted line plans through the mean environment. The blue ellipses around the path indicate the uncertainty of the system. The start and goal location are marked by an 'o' and 'x', respectively.

8. CONCLUSION

The motion planning problem for a linear, Gaussian stochastic system in an uncertain environment was formulated as an optimal control problem. The program was shown to be nonconvex and consequently only a locally optimal solution can be guaranteed. However, the allowed probability of constraint violation is ensured to be below the pre-defined threshold. This threshold is a tuning parameter which trades-off the performance and the conservativeness of the solution.

REFERENCES

Alterovitz, R., Simeon, T., and Goldberg, K. (2007). The stochastic motion roadmap: A sampling framework for planning with markov motion uncertainty. In *Proceedings of Robotics: Science and Systems*. Atlanta, GA, USA.

Blackmore, L. (2006). A probabilistic particle control approach to optimal, robust predictive control. In *Proceedings of the AIAA Guidance, Navigation, and Control Conference*. Keystone, Colorado.

Blackmore, L. and Ono, M. (2009). Convex chance constrained predictive control without sampling. In *Proceedings of the AIAA Guidance, Navigation, and Control Conference*. Chicago, Illinois.

Burns, B. and Brock, O. (2007). Sampling-based motion planning with sensing uncertainty. In *Proceedings IEEE International Conference on Robotics and Automation*. Roma, Italy.

Calafiore, G. and El Ghaoui, L. (2007). Linear programming with probability constraints. In *Proceedings of the 2007 American Control Conference*. New York, USA.

Charnes, A. and Cooper, W. (1963). Deterministic equivalents for optimizing and satisfying under chance constraints. *Operations Research*, (11), 18–39.

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

Melchior, N. and Simmons, R. (2007). Particle RRT for path planning with uncertainty. In *Proceedings IEEE International Conference on Robotics and Automation*. Roma, Italy.

Missiuro, P. and Roy, N. (2006). Adapting probabilistic roadmaps to handle uncertain maps. In *Proceedings of the IEEE International Conference on Robotics and Automation (ICRA 2006)*, 1261–1267. Orlando, FL.

Ono, M., Blackmore, L., and Williams, B.C. (2010). Chance constrained finite horizon optimal control with nonconvex constraints. In *Proceedings of the 2010 American Control Conference*. Baltimore, Maryland.

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

Prékopa, A. (1995). *Stochastic Programming*. Kluwer Academic Publishers, Dordrecht.

Prékopa, A. (1999). The use of discrete moment bounds in probabilistic constrained stochastic programming models. *Annals of Operations Research*, 85, 21–38.

van den Berg, J., Abbeel, P., and Goldberg, K. (2010). LQG-MP: Optimized path planning for robots with motion uncertainty and imperfect state information. In *Proceedings of Robotics: Science and Systems*. Zaragoza, Spain.

van Hessem, D. (2004). *Stochastic inequality constrained closed-loop model predictive control with application to chemical process operation*. Ph.D. thesis, Delft University of Technology.

Vitus, M.P., Waslander, S.L., and Tomlin, C.J. (2008). 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.