Optimal navigation for vehicles with stochastic dynamics

Shridhar K. Shah, Herbert G. Tanner and Chetan D. Pahlajani

***Abstract*—This paper presents a framework for input-optimal navigation under state constraints for vehicles exhibiting stochastic behavior. The resulting stochastic control law is implementable in real-time on vehicles with limited computational power. When control actuation is unconstrained, then convergence with probability one can be theoretically guaranteed. When inputs are bounded, the probability of convergence is quantifiable. Experimental implementation on a 5.5 g, 720 MHz processor that controls a bio-inspired crawling robot with stochastic dynamics, corroborates the design framework.**

Keywords — robot navigation; stochastic control; path integrals

I. Introduction

Miniature robots can exhibit stochastic behavior for many reasons: environmental perturbations, unmodeled compliance, ground interactions, battery charge fluctuations. Deterministic feedback navigation strategies cannot offer guarantees of convergence or obstacle avoidance when applied to systems with stochastic dynamics [3]. At the miniature scale in particular, and in conjunction with limited on-board power storage and computation capabilities, uncertainty over the system's position can be significant enough to prevent the completion of the assigned mission. Not only should uncertainty be accounted for during control design, but given that at this scale power density is limited, actuation effort must be applied sparingly.

Within an optimal control framework, uncertainty can be accounted for by either deriving worst case bounds [4]–[6], or by employing stochastic models. Comparatively, the latter can provide more flexibility and are not as conservative. One can, for example, adjust the probability that problem constraints are violated, allowing solutions that are inadmissible otherwise. However, existing methods for constrained stochastic optimal control are too computationally demanding for real-time implementation [7]–[9].

For deterministic systems, motion planning is now mature [10], and existing methods can quickly produce in an open-loop fashion waypoint sequences that connect start to goal. Systems with stochastic dynamics, however, may not be able to realize these open loop plans. When dynamics are discrete-time linear, extensions of the classical rapidly-exploring random tree (RRT) that consider probabilistic uncertainty and constraints (chance-constraints) during planning can be applied [11], [12]. The computational complexity of such chance-constrained planners [11]–[13], however, is still prohibitive for platforms on the low-end of the processor frequency range.

Work supported by ARL MAST CTA # W911NF-08-2-0004

Shah is with Mathworks Inc. shridhar.shah@mathworks.com. Tanner is the with Department of Mechanical Engineering, University of Delaware, Newark, DE, USA btanner@udel.edu

Pahlajani is with Department of Mathematics, IIT Gandhinagar, India cdpahlajani@iitgn.ac.in

Portion of this work have been previously presented at ICRA 2012 [1] and at SPIE 2013 [2]. The former conference paper dealt with systems without control multiplicative term and unbounded inputs in a linear setting. The latter conference paper presented experimental results that corroborated the theoretical predictions of the former. This paper deals with systems with control multiplicative term, considers the case of bounded inputs, and extends the theoretical study to address issues of well-posedness, existence of solutions, offer proofs of convergence, and include comparative studies.

To rein in computational complexity, receding horizon control schemes have been used on stochastic *linear discrete-time systems* [13], [14]. For linearizable nonlinear discrete-time systems, an iterative LQG [15] method offers a solution when the cost is quadratic. Particle filter approximations are used in conjunction with chance-constrained model predictive control for linear systems with probabilistic noise [13]. Other methods combine a hybrid density filter with dynamic programming [7]. Similar problems have been approached from a hybrid systems perspective in the context of a reach-avoid formulation [16], offering solutions capable of scaling up to three dimensions. In continuous-time, stochastic optimal control formulations do not fare much better. Numerical approximation methods are applied for the calculation of path integrals [17], and different applications of the formulation have been explored in conjunction with reinforcement learning [18], and risk sensitive [9] control.

This paper suggests the sequential application of pre-computed and real-time implementable locally optimal feedback strategies, that enable the system to evolve stochastically from one waypoint to the next all the way to its final destination. The approach is inspired by input-optimal exit-time stochastic optimal control formulations [19], [20]. This paper extends the aforementioned formulation to capture larger classes of systems, and adapts it to a waypoint navigation problem. It solves this problem by closing the loop through feedback controllers, shows that the resulting control system is a well defined stochastic hybrid system, and formally establishes its convergence properties. In addition to these theoretical contributions, the paper also shows how the control laws can be computed in computationally efficient ways that allow real-time application on low-speed processors, and investigates the effect of input saturation.

Compared to related stochastic optimal control frameworks, the one reported here applies directly to nonlinear systems (cf. [13]), without propagating probability densities (cf. [7]). It is developed along a similar philosophy as those in other path integral methods [9], but being built within an exit-time optimal formulation, it offers control law expressions that are not dependent on some given final time. This exit-time formulation allows solutions to be obtained at orders of magnitude faster rates (e.g., 18 vs. 5000 seconds of CPU time) compared to alternative path integral solutions [9]. After some off-line pre-computation, these solutions can be implemented in real-time on systems with up to six states.

In Section II we define a waypoint following problem, which can be mathematically encoded as an exit-time stochastic optimal control problem. Section III demonstrates that the optimal control problem of transitioning from waypoint to waypoint is associated with a partial differential equation (PDE), which can be solved numerically by leveraging the Feynman-Kac formula. In Section IV, we show that the resulting closed loop system is essentially a special case of a Markov string with well-defined solutions, and we prove that its convergence can be guaranteed. Section V assesses the performance of the closed loop system in terms of computational complexity and optimality, and demonstrates the scheme's applicability to miniature vehicles driven by low-end processors.

II. Problem Statement

Consider the motion of vehicles with dynamics that can be represented as a stochastic process:

$$\begin{aligned} \mathrm{dq} &= b(\mathrm{q})\,\mathrm{d}t + G(\mathrm{q})\big[u(i,\mathrm{q})\,\mathrm{d}t + \Sigma(\mathrm{q})\,\mathrm{d}W\big] \\ \mathrm{q}(0) &= \mathrm{q}_0\ , \end{aligned} \tag{1}$$

where $\mathrm{q} \in \mathbb{R}^n$ is the state, $b : \mathbb{R}^n \to \mathbb{R}^n$ is the drift term, $G : \mathbb{R}^n \to \mathbb{R}^{n\times m}$ is the matrix of control vector fields, $i \in \mathcal{I} \triangleq \{0, 1, \dots, N\}$

is a switching index, $u : \mathcal{I} \times \mathbb{R}^n \to \mathbb{R}^m$ is the control input, and $\Sigma : \mathbb{R}^n \to \mathbb{R}^{m \times m}$ is the diffusion matrix. Let $W = \{W(t), \mathcal{F}_t : 0 \le t < \infty\}$ be an m-dimensional Wiener process on the probability space $(\Omega, \mathcal{F}, \mathbb{P})$, where Ω is the sample space, $\mathcal{F}$ is a σ-algebra on Ω, $\mathbb{P}$ is the probability measure and $\{\mathcal{F}_t : t \ge 0\}$ is a right-continuous filtration [21]. Assume that $b(\mathrm{q})$, $G(\mathrm{q})$, $\Sigma(\mathrm{q})$, and $\Sigma^{-1}(\mathrm{q})$ are bounded and Lipschitz continuous—standard assumptions for this type of stochastic processes [21].

Process (1) evolves in a domain $\mathcal{P}$, which is a subset of a larger bounded domain $\mathcal{S} \subset \mathbb{R}^n$, called the workspace. Within $\mathcal{S}$, there is a closed set $\overline{\mathcal{O}} \subset \mathcal{S}$ ($^-$ denotes closure) which represents obstacles to be avoided. Thus $\mathcal{P} \triangleq \mathcal{S} \setminus \overline{\mathcal{O}}$ is the system's *free workspace*. The goal is to steer $\mathrm{q}(t)$ to a goal set $\overline{\mathcal{G}} \subset \mathcal{P}$ in finite time, avoiding $\partial\mathcal{P}$ (∂ denotes boundary), with minimal control effort. We assume $\partial\mathcal{G}$ and $\partial\mathcal{P}$ are disjoint, that the free workspace is connected, and that all boundaries $\partial\mathcal{S}$, $\partial\mathcal{O}$, $\partial\mathcal{P}$ and $\partial\mathcal{G}$, are twice continuously differentiable (i.e., C^2). From an implementation viewpoint, we want to realize this solution on computational platforms less powerful than currently available cellular devices.

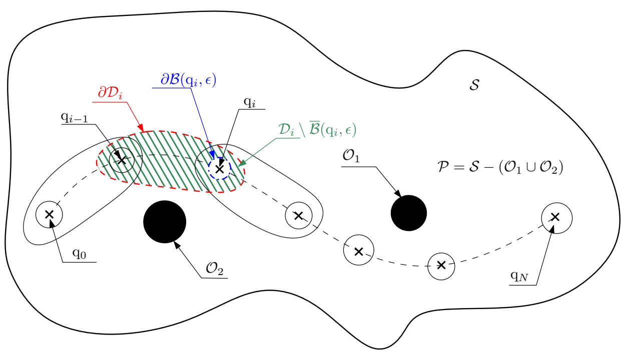


Fig. 1: The stochastic system evolves within the local domains $\mathcal{D}_i \setminus \overline{\mathcal{G}}_i$ (hashed region–here, $\overline{\mathcal{G}}_i \triangleq \overline{\mathcal{B}}(\mathrm{q}_i, \epsilon)$). The waypoint sequence $\{\mathrm{q}_i\}$ is marked with crosses over a continuous curve, the obstacles $\mathcal{O}_j$ are solid disks, the boundaries $\partial\mathcal{G}_i = \partial\mathcal{B}(\mathrm{q}_i, \epsilon)$ are shown in dashed blue contours, and boundaries $\partial\mathcal{D}_i$ appear as dashed red closed curves.

More technically, we require that any system sample path connects successively elements of a sequence of neighborhoods called *goal sets* $\mathcal{G}_i, i = 0, \ldots, N$, which are centered around some given *waypoints*. In this sequence, $\mathcal{G}_0$ is a neighborhood of the system's initial position, and $\mathcal{G}_N$ is a neighborhood of its desired destination. The transition between $\mathcal{G}_{i-1}$ and $\mathcal{G}_i$, occurring in the time interval $[t_{i-1}, t_i]$, should be optimal in the sense that the functional

$$J(i, \mathrm{q}(t_{i-1})) = \mathbb{E}^{\mathrm{q}(t_{i-1})}\Big[\int_{t_{i-1}}^{t_i} L\big(\mathrm{q}(s), u(i, \mathrm{q}(s))\big)\,\mathrm{d}s + \Phi\big(\mathrm{q}(t_i)\big)\Big]$$

is minimized. Above, $\mathbb{E}$ denotes expectation with the superscript specifying the initial condition of the sample paths, $L\big(\mathrm{q}, u\big) \triangleq \frac{1}{2}u^\intercal(\mathrm{q})\big(\Sigma(\mathrm{q})\Sigma^\intercal(\mathrm{q})\big)^{-1}u(\mathrm{q})$, and $\Phi(\mathrm{q}) : \mathbb{R}^n \to \mathbb{R}_+$ is a terminal cost function. Time instant t_i is an *exit time*: the first time that the state hits $\mathcal{G}_i$. With $\mathbb{P}$ denoting probability, the solution is subject to the following constraints:

$$\begin{aligned}
&\mathbb{P}^{\mathrm{q}(t_{i-1})}\left[\mathrm{q}(t_i) \in \partial\mathcal{G}_i\right] = 1 \ , \\
&\mathbb{P}^{\mathrm{q}(t_{i-1})}\left[\mathrm{q}(t) \in \mathcal{D}_i\right] = 1, \forall t \in [t_{i-1}, t_i] \ ,
\end{aligned}$$

where $\mathcal{D}_i$ can be any obstacle-free bounded domain properly containing both $\mathcal{G}_{i-1}$ and $\mathcal{G}_i$. The former constraint requires *almost-sure convergence*: that the system will reach its destination with probability one. The latter is an *almost-sure safety* constraint: that the system's probability of colliding with obstacles is zero.

III. Switched stochastic optimal control

The sequence of waypoints can be produced in a number of different ways (e.g., RRTs, potential, or harmonic fields, or other specialized methods [22]) using the deterministic part (drift) of (1)—it will not be discussed further here. Having a sequence of sets $\{\mathcal{G}_i\}_{i=0}^{N}$, we develop a stochastic optimal controller that can ensure that (1) transitions from $\overline{\mathcal{G}}_{i-1} \subset \mathcal{D}_i$ to $\overline{\mathcal{G}}_i$ in finite time.

Practically, $\bigcup_i \mathcal{D}_i$ defines a "tunnel" in the obstacle-free space that should contain the system's trajectories. Reducing the width of this tunnel will keep trajectories close to waypoints at the expense of control actuation. On the other hand, increasing the size of the tunnel, allows the system to use more of the available free space, using actuation less frequently to steer away from boundaries. If, for instance, numerical considerations force $\mathcal{D}_i$ to be spheres, then one chooses the maximal radius R_i possible that ensures $\mathcal{D}_i \cap \overline{\mathcal{O}} = \varnothing$.

Convergence to $\overline{\mathcal{G}}_i$, and satisfaction of state constraints, is in theory guaranteed with probability one. Recall (1):

$$\mathrm{d}\mathrm{q}(t) = b\big(\mathrm{q}(t)\big)\,\mathrm{d}t + G\big(\mathrm{q}(t)\big)\Big[u\big(i, \mathrm{q}(t)\big)\,\mathrm{d}t + \Sigma\big(\mathrm{q}(t)\big)\,\mathrm{d}W(t)\Big] \ ,$$

take as initial condition $\mathrm{q}(t_{i-1}) \in \overline{\mathcal{G}}_{i-1}$, and define

$$V(i, \mathrm{q}) \triangleq \min_{u(i,\mathrm{q})} \mathbb{E}^{\mathrm{q}(t_{i-1})}\Big[\int_{t_{i-1}}^{t_i} L\big(\mathrm{q}(s), u(i, \mathrm{q}(s))\big)\,\mathrm{d}s + \Phi\big(\mathrm{q}(t_i)\big)\Big] \ . \tag{2}$$

The associated Hamilton-Jacobi-Bellman (HJB) equation is [23]

$$\min_{u(i,\mathrm{q})} \Big\{\mathcal{A}V(i, \mathrm{q}) + L\big(\mathrm{q}(s), u(i, \mathrm{q}(s))\big)\Big\} = 0 \tag{3}$$

where $\mathcal{A}$ is the second-order partial differential operator

$$\begin{aligned}
\mathcal{A}V(i, \mathrm{q}) \triangleq\ & \partial_{\mathrm{q}}V(i, \mathrm{q})\ \big(b(\mathrm{q}) + G(\mathrm{q})u(i, \mathrm{q})\big) \\
&+ \frac{1}{2}\mathrm{tr}\left[\partial_{\mathrm{qq}}V(i, \mathrm{q})\ G(\mathrm{q})\Sigma(\mathrm{q})\Sigma^\intercal(\mathrm{q})G^\intercal(\mathrm{q})\right] \ ,
\end{aligned}$$

in which tr stands for matrix trace, $\partial_{\mathrm{q}} \equiv \frac{\partial}{\partial \mathrm{q}}$ is gradient row vector, and $\partial_{\mathrm{qq}} \equiv \frac{\partial^2}{\partial \mathrm{q}^2}$ is the operator generating the Hessian of a scalar function. Expanding (3) we get

$$\begin{aligned}
\min_{u(i,\mathrm{q})} \Big\{&\partial_{\mathrm{q}}V(i, \mathrm{q})b(\mathrm{q}) + \partial_{\mathrm{q}}V(i, \mathrm{q})\,G(\mathrm{q})u(i, \mathrm{q}) \\
&+ \frac{1}{2}\mathrm{tr}\left[\partial_{\mathrm{qq}}V(i, \mathrm{q})\ G(\mathrm{q})\Sigma(\mathrm{q})\Sigma^\intercal(\mathrm{q})G^\intercal(\mathrm{q})\right] \\
&+ \frac{1}{2}u^\intercal(i, \mathrm{q})\ \big(\Sigma(\mathrm{q})\Sigma^\intercal(\mathrm{q})\big)^{-1}u(i, \mathrm{q})\Big\} = 0 \ .
\end{aligned} \tag{4}$$

The optimal control law $u^*(i, \mathrm{q})$ that satisfies (4) is now

$$u^*(i, \mathrm{q}) = -\Sigma(\mathrm{q})\Sigma^\intercal(\mathrm{q})\,G^\intercal(\mathrm{q})\ \partial_{\mathrm{q}}V^\intercal(i, \mathrm{q}) \ . \tag{5}$$

Substituting (5) in (4) yields

$$\begin{aligned}
\partial_{\mathrm{q}}V(i, \mathrm{q})b(\mathrm{q}) &- \frac{1}{2}\partial_{\mathrm{q}}V(i, \mathrm{q})G(\mathrm{q})\Sigma(\mathrm{q})\Sigma^\intercal(\mathrm{q})\,G^\intercal(\mathrm{q})\partial_{\mathrm{q}}V^\intercal(i, \mathrm{q}) \\
&+ \frac{1}{2}\mathrm{tr}\left[\partial_{\mathrm{qq}}V(i, \mathrm{q})\ G(\mathrm{q})\Sigma(\mathrm{q})\Sigma^\intercal(\mathrm{q})G^\intercal(\mathrm{q})\right] = 0 \ .
\end{aligned} \tag{6}$$

Using the logarithmic transformation $V(i, \mathrm{q}) = -\log g(i, \mathrm{q})$ in (6) we get the PDE [20]

$$\partial_{\mathrm{q}}g(i, \mathrm{q})\,b(\mathrm{q}) + \frac{1}{2}\mathrm{tr}\left[G(\mathrm{q})\Sigma(\mathrm{q})\ \partial_{\mathrm{qq}}g(i, \mathrm{q})\ \Sigma^\intercal(\mathrm{q})G^\intercal(\mathrm{q})\right] = 0 \tag{7}$$

with boundary condition

$$g(i, \mathrm{q}) = \exp\Big(-\Phi\big(\mathrm{q}(t_i)\big)\Big), \ \ \mathrm{q} \in \partial\mathcal{D}_i \cup \partial\mathcal{G}_i \ .$$

We now guarantee that the system does not exit from a specific portion of the boundary by setting

$$\Phi \triangleq \begin{cases} 0 & \text{on} \quad \partial\mathcal{G}_i \\ \infty & \text{on} \quad \partial\mathcal{D}_i \end{cases} \implies g(i, \mathrm{q}) = \begin{cases} 1 & \text{on} \quad \partial\mathcal{G}_i \\ 0 & \text{on} \quad \partial\mathcal{D}_i \end{cases} .$$

Analytic solutions of (7) are in general not possible for nonlinear systems. However, the Feynman-Kac formula [21, Lemma 5.7.4] relates this PDE to an equivalent stochastic differential equation (SDE), and expresses the solution of (7) in the form

$$g(i, \mathrm{q}) = \mathbb{P}^{\mathrm{q}} \left[\zeta(\tau) \in \partial\mathcal{G}_i\right] \,, \tag{8}$$

where $\zeta(t)$ is the (uncontrolled) Markov process

$$\mathrm{d}\zeta(t) = b\big(\zeta(t)\big)\,\mathrm{d}t + G\big(\zeta(t)\big)\,\Sigma\big(\zeta(t)\big)\,\mathrm{d}W(t); \quad \zeta(0) = \mathrm{q} \tag{9}$$

evolving on the same bounded domain $\mathcal{D}_i \setminus \overline{\mathcal{G}}_i \subset \mathbb{R}^n$, with τ being the exit time from $\mathcal{D}_i \setminus \overline{\mathcal{G}}_i$ for initial condition $\mathrm{q} \in \mathcal{D}_i \setminus \overline{\mathcal{G}}_i$. Then (8) suggests that $g(i, \mathrm{q})$ is in fact the probability that a sample path of (9) from initial condition q hits the boundary $\partial\mathcal{G}_i$ before hitting $\partial\mathcal{D}_i$. If the optimal control (5) exists, then the infinite penalty on exiting the wrong boundary is equivalent to a constraint

$$\mathbb{P}^{\mathrm{q}(t_{i-1})}\big[\mathrm{q}(t_i) \in \partial\mathcal{D}_i\big] = 0 \Leftrightarrow \mathbb{P}^{\mathrm{q}(t_{i-1})}\big[\mathrm{q}(t_i) \in \partial\mathcal{G}_i\big] = 1 \,. \tag{10}$$

The proof of the above statement can be constructed along the same steps as that in [19]. We ensure that the exit time is finite by assuming that for some $l \in \{1, \dots, m\}$

$$\min_{\mathrm{q} \in \mathcal{D}_i \setminus \overline{\mathcal{G}}_i} A_{ll}(\mathrm{q}) > 0 \tag{11}$$

where $A = G\ \Sigma\Sigma^{\intercal}\ G^{\intercal}$. This is in fact a mild assumption on non-degeneracy of noise, which yields $\mathbb{E}^{\mathrm{q}(t_{i-1})}[t_i] < \infty$ [21, Lemma 7.4]. With optimal controller (5), which can be computed by either solving (7) for $g(i, \mathrm{q})$ analytically, or numerically simulating (9) and then estimating (8), we can close the loop around (1) in a way that makes this system transition from $\mathcal{G}_{i-1}$ to $\mathcal{G}_i$ in finite time. Doing so iteratively for $i = 1, \dots, N$ allows the system to navigate from its initial configuration, while staying always inside some domain $\mathcal{D}_i$, and therefore avoiding obstacles and converging with probability one.

IV. A Stochastic Hybrid System

A. Definition

Closing the loop around (1) using (5) gives rise to a well-defined special case of a stochastic hybrid system [24]. To see how our closed loop system falls within thegereral stochastic hybrid system (GSHS) class, define the hybrid state as (i, q). In this pair, $\mathrm{q} \in \mathbb{R}^n$ and $i \in \{0, 1, 2, ..., N\}$ are the continuous and discrete components of the hybrid state, respectively. Now the components of the stochastic hybrid system can be defined as follows:

- Continuous dynamics: The SDE (1).
- Discrete dynamics: The discrete state i evolves by means of state-triggered forced transitions $i \to i+1$, occurring at *stopping time* $t_i \triangleq \inf\{t > t_{i-1} \mid \mathrm{q}(t) \notin \mathcal{D}_i \setminus \overline{\mathcal{G}}_i\}$. Note that due to the set of discrete states being finite, and the discrete transition map being a bijection, there can only be a finite number of discrete transitions and the system cannot exhibit Zeno behavior.
- Reset condition: The reset map for the continuous states is simply the identity.

The solution of (1) for $i \in \{1, \dots, N\}$ now becomes a collection of Markov processes truncated at (their) exit time, which can be represented as a *Markov string*. A Markov string is a hybrid state jump Markov process [24]. Given the existence of solutions for each SDE (1) for fixed i [19], and due to N being finite, the solutions for the closed-loop stochastic hybrid system are well-defined [24].

B. Convergence Properties

We can now show that the closed loop stochastic hybrid system generates sample paths that converge to a neighborhood of the origin. Henceforth, without loss of generality, this origin is assumed to be in the interior of the system's destination, or target set.

Proposition 1. *Consider the switched stochastic system* (1) *in an open bounded domain $\mathcal{P} \subset \mathbb{R}^n$, where $i \in \{1, \dots, N\}$ is the switching index, and $W(t)$ is a Wiener process. Let $K(\cdot)$ be a $\mathcal{C}^2$ control Lyapunov function for the deterministic system $\dot{x} = b(x) + G(x)u$, defined in the closure of the bounded domain $\mathcal{P}$ which contains the origin. If for every solution $\mathrm{q}(t)$ of* (1) *there exist*

- *a class-$\mathcal{K}$ function η on $\mathcal{P}$, together with a sequence of points $\{\mathrm{q}_i\}_{i=0}^{N} \in \mathcal{P}$ satisfying*
$$\max_{a \in \overline{\mathcal{G}}_i}\{K(a)\} - \min_{b \in \overline{\mathcal{G}}_{i-1}}\{K(b)\} \leq -\eta(\|\mathrm{q}_{i-1}\|) \tag{12}$$
- *and bounded domains $\mathcal{D}_i \subset \mathcal{P}$ that satisfy*
$$\overline{\mathcal{G}}_{i-1} \subset \mathcal{D}_i \,, \overline{\mathcal{G}}_i \subset \mathcal{D}_i \,, \overline{\mathcal{G}}_{i-1} \cap \overline{\mathcal{G}}_i = \varnothing \,, \tag{13}$$

then $\forall i \in \{0, \dots, N\}$, $\mathcal{D}_i$ are positively invariant and the closed-loop switched stochastic system (1)–(5) *converges to an ϵ-neighborhood of origin in finite time with probability one.*

Proof: Set $\hat{\mathrm{q}}(t_0) = \mathrm{q}_0$, and construct a path Q_T of finite length, having a partition with $N-1$ segments such that there is a segment that links point q_{i-1} to q_i, for all $i = 1, \dots, N$. Given that the initial condition for (1)–(5) is inside $\mathcal{D}_0$, the invariance of $\mathcal{D}_i$ for every $i \in \{0, \dots, N\}$ follows from (10). Since now the bounded domain $\mathcal{D}_1$ satisfies $\mathcal{D}_1 \subset \mathcal{P}$ and (13), the application of control law (5) ensures that for all $\mathrm{q}(t_0) \in \mathcal{G}_0$, $\mathbb{P}^{\mathrm{q}(t_0)}[\mathrm{q}(t_1) \in \partial\mathcal{G}_1] = 1$, that is, the state at time t_1 is in $\mathcal{G}_1$ almost surely (see [19]). Condition (11) ensures that the time that this happens is finite. Now, if controller $u(k, \mathrm{q})$ is applied iteratively, at some time t_k, $\mathrm{q}(t_k) \in \partial\mathcal{G}_k$. As $\overline{\mathcal{G}}_k \subset \mathcal{D}_{k+1}$ and given (13), there exists a controller $u(k+1, \mathrm{q})$ to steer the state to the next goal set $\overline{\mathcal{G}}_{k+1}$. Given now that $\mathcal{D}_{k+1}$ also satisfies $\mathcal{D}_{k+1} \subset \mathcal{P}$ and (13), the law (5) gives $\mathbb{P}^{\mathrm{q}(t_k)}[\mathrm{q}(t_{k+1}) \in \partial\mathcal{G}_{k+1}] = 1$ with $\mathbb{E}^{\mathrm{q}(t_k)}[t_{k+1}] < \infty$. Inductively, since $\mathcal{G}_N := \overline{\mathcal{B}}(0, \epsilon)$, the proof is completed. ■

V. Performance Assessment

This section reports on performance evaluation results for the closed loop stochastic navigation strategy, and compares the solution with existing methodologies developed along a similar philosophy. Our assessment indicates that the reported solution (i) compares favorably computationally with alternatives, (ii) can be successfully implemented experimentally in real-time on compliant miniature multi-legged robots controlled by 720 MHz processors, and (iii) the trade-off established between optimality and computational speed is such that the former is not significantly affected.

A. Comparison with existing solutions

Take the particle in half space example of [9]:

$$\mathrm{d}X_t^1 = u\,\mathrm{d}t + \sigma\,\mathrm{d}W \;\;, \quad X_0^1 = 1 \;\;,$$

with $\sigma^2 = 10$, and u unconstrained. The optimization objective in [9] is to minimize over a time interval of $[0, T]$ with $T = 10$ s, the expectation

$$\mathbb{E}\Big[50(X_T^1 - 1)^2 + \int_0^T V(X_s^1) + 0.5u(X_s^1)^2\,\mathrm{d}s\Big]$$

where $V(x) = \infty$ if $x \leq 0$, and zero otherwise. The system starts from 1 and crosses 1 again in $T = 10$ s, while minimizing the above expectation. This is a finite-time optimization problem.

The formulation in this paper gives rise to an infinite-horizon optimal control problem. In addition, the objective is to hit a set, rather than to cross a threshold value from either direction, and our constraints are bidirectional, not unidirectional ($x > 0$) as in [9]. Finally, the solution of this paper offers formal guarantees of convergence to the desired final state. Having stated these differences, we proceed to formulate an optimal control problem in our framework trying to stay as close as possible to that of [9]. We keep the same initial condition and assign $\mathcal{D} := (0, 4)$ and $\mathcal{G} := [0.9, 1.1]$. Whereas in [9, Fig. 4] the computation time required for the approximated optimal controller to yield a cost in the neighborhood of the analytically computed value is close to 5 000 seconds, our optimal solution is obtained in just 17.32 seconds.

B. Optimality gap

In this section we investigate the difference in terms of cost, between a globally optimal solution from initial configuration q_0 all the way to its destination, and the one generated by the stochastic hybrid system steered by (1) along the series of waypoints $\{\mathrm{q}_i\}_{i=1}^N$. Specifically, we want to see how suboptimal the solution can be when the trajectory is constrained to (i) pass close to the waypoints, and (ii) stay within the domains $\mathcal{D}_i$.

For the comparison to be meaningful, we generate the series of waypoints in a suboptimal manner, by solving a receding horizon optimization problem on the deterministic part $\dot{q} = u$ of the dynamics of a stochastic single integrator

$$\mathrm{dq} = u(i, \mathrm{q})\,\mathrm{d}t + \Sigma(\mathrm{q})\,\mathrm{d}W; \;\; \mathrm{q}(0) = \mathrm{q}_0 \;, \qquad (14)$$

required to navigate in a two-dimensional workspace that includes two spherical obstacles (Fig. 2). In (14), $\mathrm{q} = [x\ y]^\intercal \in \mathbb{R}^2$ is the state of the system, and $W(t)$ is a 2-dimensional Wiener process. For the generation of the waypoint sequence (solid red disks in Fig. 2), we follow [25]. The time history of the control inputs that steer (14) along the sample path of Fig. 2 is shown in Fig. 3(a).

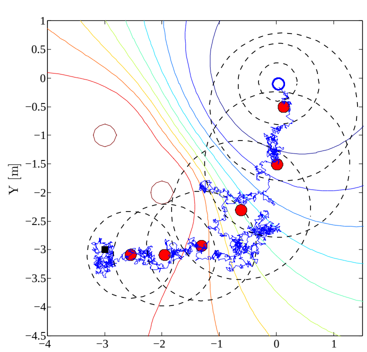
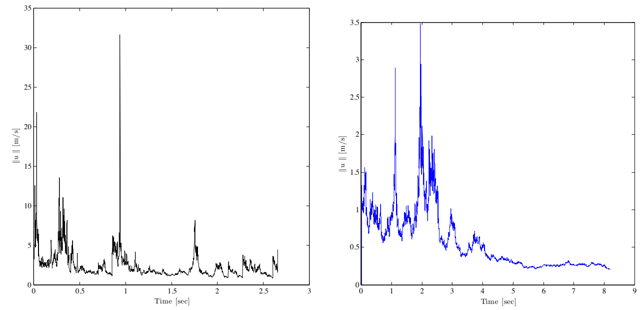


Fig. 2: Navigation of (14) in an obstacle environment



(a) Control effort in Fig. 2 (b) Optimal inputs without obstacles

Fig. 3: Comparison of input effort between locally optimal waypoint navigation and globally optimal unconstrained control

We now evaluate analytically and implement a globally optimal controller based on (5), for a single domain of the same radius as the cluttered one of Fig. 2 but without any obstacles, for simplicity—this simplification is likely to yield more optimistic cost estimates, which are nevertheless just as useful for comparison purposes. We generated 50 sample paths of this globally optimal control law in (14) to obtain cost statistics. The control input history for a representative sample path is shown in Fig. 3(b).

A visual comparison of the two representative control histories in Fig. 3 reveals that restraining the system sample paths inside the domains $\mathcal{D}_i$ increases the magnitude of control inputs utilized—as much as tenfold, in the particular example. On the other hand, the time it takes the system to converge is much shorter in the waypoint navigation case; almost three times shorter for Fig. 3. This is due to the optimal controller (5) letting the system drift on its own for extended periods of time, as long as it does not come close to the workspace boundaries. As a result, the integral of the control effort over *time* as measured by (2) is significantly increased. In some sense, it appears that the waypoints have the potential to funnel the system to its destination fast enough to counterweight the effect of the local optimization. The cost of the 50 sample paths produced by the globally optimal unconstrained controller is $5.9 \pm 0.9\ \mathrm{m}^2/\mathrm{s}$. The cost for the sample path shown in Fig. 2 is $6.2\ \mathrm{m}^2/\mathrm{s}$.

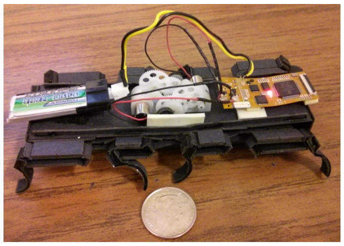
C. Experimental implementation

This section discusses results obtained from an experimental evaluation on a robotic system with stochastic kinematics (Fig. 4(a)). The implementation demonstrates that the application of the reported controller is relatively inexpensive computationally, and feasible for an Overo Fire COM board with 720 MHz processor and 500 MB RAM.

The stochasticity in the motion of the robot in Fig. 4(a) comes primarily due to the effect of ground interactions, which is relatively significant for a platform of that size factor [26]. The deterministic kinematics of crawling robots with similar differential steering but larger size than that of the OctoRoACH have been reasonably approximated by a unicycle model [27]. We therefore hypothesize that a reasonable kinematic model for the OctoRoACH of Fig. 4(a) would be a Dubin's car model with a stochastic extension:

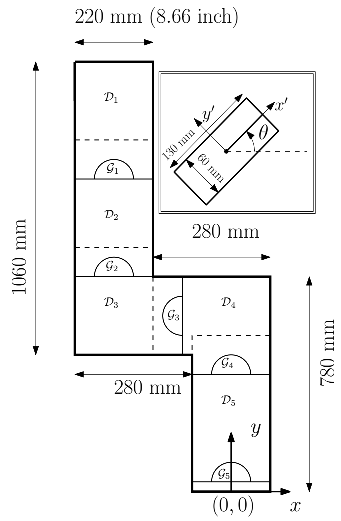
$$\underbrace{\begin{bmatrix}\mathrm{d}x\\ \mathrm{d}y\\ \mathrm{d}\theta\end{bmatrix}}_{\mathrm{dq}} = \underbrace{\begin{bmatrix}v\cos\theta\\ v\sin\theta\\ 0\end{bmatrix}}_{b(\mathrm{q})}\mathrm{d}t + \underbrace{\begin{bmatrix}0\\0\\1\end{bmatrix}}_{G(\mathrm{q})}\left(\underbrace{\omega}_{u(i,\mathrm{q})}\,\mathrm{d}t + \Sigma\,\mathrm{d}W(t)\right), \qquad (15)$$

where linear velocity v and variance Σ are constants, experimentally



(a) The OctoRoACH robot

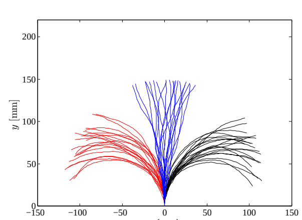
(b) The workspace



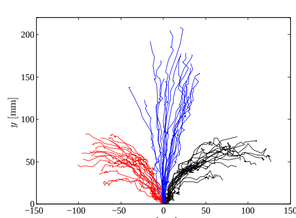
(c) Domains and dimensions

Fig. 4: Experimental scenario. The small crawling robot in (a) is to navigate along a narrow corridor shown in (b), outside which it falls and fails the mission. A diagram of the robot's workspace with dimensions and assigned coordinate systems in shown in (c), where the domains $\mathcal{D}_i$, and the goal sets $\mathcal{G}_i$ for $i = 1, \ldots, 5$ are sketched. The boundaries of these sets are approximated by C^2 functions.

estimated as 0.02 m/s and 0.25 respectively.[1] Figure 5 suggests that this model hypothesis is valid.



(a) Stochastic Dubin's car



(b) OctoRoACH open-loop paths

Fig. 5: Stochastic Dubin's car simulated paths (left) and experimental OctoRoACH paths (right)

For the rectangular (with rounded corners) domain of Fig. 4(c), the

[1]An alternative formal method for estimating noise intensity from experimental data is reported in [28].

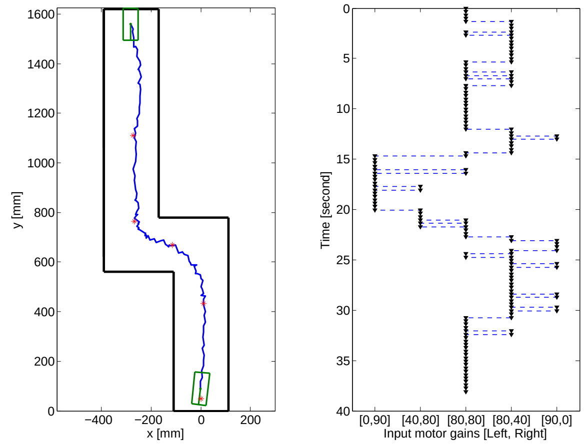


Fig. 6: Experimental path. Left: the observed path of the robot, in blue, along the waypoints ($*$) superimposed on the workspace, in black. Right: input motor gains applied.

numerical solution of PDE (7) is obtained off-line, and uploaded on the robot's processor in order for the control inputs to be calculated in real-time. Unless the shape of the domains needs to be changed, the PDE does not need to be resolved; the robot can use the pre-computed solution in workspaces with different geometry, as long as the free space admits concatenations of any *scaled* versions of these domains. Figure 6 shows a path and the input motor gains used during the experiment. Although there can be additional sources of energy dissipation during execution, e.g. communication, computation, PWM motor driving, etc., this optimization targets solely the portion of power that goes toward actuator utilization.

VI. Conclusions

Miniature robotic vehicles that exhibit stochastic dynamics require special control techniques for navigation in cluttered environments. Stochasticity needs to be taken into account in control design. Whereas stochastic optimal control methods are notoriously expensive computationally, there exist relaxations that allow efficient solutions with theoretical guarantees of safety and convergence, that are implementable on processors at the low end of the frequency scale without sacrificing too much in terms of performance. This paper outlines one such solution. It involves the off-line computation of the value function of a HJB equation for robot domains of predefined shape surrounding the robot. The solution can then be "scaled" and reused in different problem instantiations without the need for additional computations. Nonlinear stochastic models with up to six states are well within the applicability range of the reported control design approach, based on available mobile computing power.

References

[1] S. Shah, C. Pahlajani, N. Lacock, and H. Tanner, "Stochastic receding horizon control for robots with probabilistic state constraints," in *IEEE International Conference on Robotics and Automation*, Minnesota, USA, May 2012, pp. 2893–2898.

[2] S. K. Shah and H. G. Tanner, "Stochastic receding horizon control: Application to an octapedal robot," in *Proceedings of the SPIE Defense, Security, and Sensing*, 2013, pp. 87 251D–87 251D–9.

[3] S. Shah, C. Pahlajani, and H. Tanner, "Probability of success in stochastic robot navigation with state feedback," in *IEEE/RSJ International Conference on Intelligent Robots and Systems*, 2011, pp. 3911–3916.

[4] D. Ramírez, T. Alamo, E. Camacho, and D. M. de la Peña, "Min-max MPC based on a computationally efficient upper bound of the worst case cost," *Journal of Process Control*, vol. 16, no. 5, pp. 511–519, 2006.

[5] D. DeHaan and M. Guay, *Model Predictive Control*, T. Zheng, Ed. Vienna, Austria: Sciyo, 2010.

[6] D. Marruedo, T. Alamo, and E. Camacho, "Input-to-state stable mpc for constrained discrete-time nonlinear systems with bounded additive uncertainties," in *IEEE Conference on Decision and Control*, Nevada, USA, December 2002, pp. 4619 – 4624.

[7] F. Weissel, M. Huber, and U. Hanebeck, "A nonlinear model predictive control framework approximating noise corrupted systems with hybrid transition densities," in *IEEE Conference on Decision and Control*, Louisiana, USA, December 2007, pp. 3661 –3666.

[8] L. Blackmore, M. Ono, A. Bektassov, and B. Williams, "A probabilistic particle-control approximation of chance-constrained stochastic predictive control," *IEEE Transactions on Robotics*, vol. 26, no. 3, pp. 502 –517, 2010.

[9] B. van den Broek, W. Wiegerinck, and H. Kappen, "Stochastic optimal control of state constrained systems," *International Journal of Control*, vol. 84, no. 3, pp. 597–615, 2011.

[10] S. M. LaValle, *Planning Algorithms*. Cambridge, U.K.: Cambridge University Press, 2006.

[11] B. Luders, M. Kothari, and J. P. How, "Chance constrained RRT for probabilistic robustness to environmental uncertainty," in *AIAA Guidance, Navigation, and Control Conference*, Toronto, Canada, August 2010.

[12] G. Kewlani, G. Ishigami, and K. Iagnemma, "Stochastic mobility-based path planning in uncertain environments," in *IEEE/RSJ International Conference on Intelligent Robots and Systems*, October 2009, pp. 1183–1189.

[13] L. Blackmore, M. Ono, and B. Williams, "Chance-constrained optimal path planning with obstacles," *IEEE Transactions on Robotics*, vol. 27, no. 6, pp. 1080 –1094, 2011.

[14] P. Hokayem, E. Cinquemani, D. Chatterjee, F. Ramponi, and J. Lygeros, "Stochastic receding horizon control with output feedback and bounded controls," *Automatica*, vol. 48, no. 1, pp. 77 – 88, 2012.

[15] E. Todorov and W. Li, "A generalized iterative LQG method for locally-optimal feedback control of constrained nonlinear stochastic systems," in *American Control Conference*, Oregon, USA, June 2005, pp. 300–306.

[16] S. Summers and J. Lygeros, "Verification of discrete time stochastic hybrid systems: A stochastic reach-avoid decision problem," *Automatica*, vol. 46, no. 12, pp. 1951 – 1961, 2010.

[17] H. J. Kappen, "Path integrals and symmetry breaking for optimal control theory," *Journal of Statistical Mechanics: Theory and Experiment*, vol. 2005, no. 11, p. P11011, 2005.

[18] E. Theodorou, F. Stulp, J. Buchli, and S. Schaal, "Iterative path integral stochastic optimal control for learning robotic tasks," in *18th IFAC World Congress*, Milano, Italy, August 2011.

[19] M. Day, "On a stochastic control problem with exit constraints," *Applied Mathematics and Optimization*, vol. 6, pp. 181–188, 1980.

[20] W. H. Fleming, "Exit probabilities and optimal stochastic control," *Applied Mathematics and Optimization*, vol. 4, pp. 329–346, 1977.

[21] I. Karatzas and S. E. Shreve, *Brownian Motion and Stochastic Calculus*, 2nd ed. Springer, 1991.

[22] K. H. Low and S. Bai, "Terrain-evaluation-based motion planning for legged locomotion on irregular terrain," *Advanced Robotics*, vol. 17, no. 8, pp. 761–778, 2003.

[23] W. H. Fleming and H. M. Soner, *Controlled Markov Processes and Viscosity Solutions*, 2nd ed. Springer, 2005.

[24] M. Bujorianu and J. Lygeros, "Toward a general theory of stochastic hybrid systems," in *Stochastic Hybrid Systems: Theory and Safety Critical Applications*, ser. Lecture Notes in Control and Information Sciences, 2006, vol. 337, pp. 3–30.

[25] H. Tanner and J. Piovesan, "Randomized receding horizon navigation," *IEEE Transactions on Automatic Control*, vol. 55, no. 11, pp. 2640 – 2644, 2010.

[26] A. Pullin, N. Kohut, D. Zarrouk, and R. Fearing, "Dynamic turning of 13 cm robot comparing tail and differential drive," in *IEEE International Conference on Robotics and Automation*, Minnesota, USA, May 2012, pp. 5086–5093.

[27] D. Panagou and H. Tanner, "Modeling of a Hexapod Robot; Kinematic Equivalence to a Unicycle," University of Delaware, Department of Mechanical Engineering, Tech. Rep., 04 2009.

[28] K. Karydis, I. Poulakakis, and H. G. Tanner, "Probabilistic validation of a stochastic kinematic model for an eight-legged robot," in *Proceedings of the IEEE International Conference on Robotics and Automation*, 2013, pp. 2562–2567.