Solving the Hamilton-Jacobi-Bellman Equation for a Stochastic System with State Constraints

PER RUTQUIST

TORSTEN WIK

CLAES BREITHOLTZ

Department of Signals and Systems
Division of Automatic Control, Automation and Mechatronics
CHALMERS UNIVERSITY OF TECHNOLOGY
Gothenburg, Sweden, 2014
Report No. R007/2014
ISSN 1403-266X

Solving the Hamilton-Jacobi-Bellman equation for a stochastic system with state constraints[*]

Per Rutquist, Torsten Wik and Claes Breitholtz

March 21, 2014

Abstract

We present a method for solving the Hamilton-Jacobi-Bellman (HJB) equation for a stochastic system with state constraints. A variable transformation is introduced which turns the HJB equation into a combination of a linear eigenvalue problem, a set of partial differential equations (PDE:s), and a point-wise equation. For a fixed solution to the eigenvalue problem, the PDE:s are linear and the point-wise equation is quadratic, indicating that the problem can be solved efficiently using an iterative scheme.

As an example, we numerically solve for the optimal control of a Linear Quadratic Gaussian (LQG) system with state constraints. A reasonably accurate solution is obtained even with a very small number of collocation points (three in each dimension), which suggests that the method could be used on high order systems, mitigating the curse of dimensionality.

1 Introduction

A stochastic optimal control problem for a system with state constraints can be solved by finding the solution to the corresponding Hamilton-Jacobi-Bellman (HJB) equation [1]. However, the resulting partial differential equation (PDE) is nonlinear and difficult to solve as it will in general also have infinite boundary conditions. For systems where both the input and the (white noise) disturbances enters affinely, a logarithmic transformation [4] can be used to achieve an exact linearization of the HJB equation [8, 9, 10, 14].

[*] This paper has been submitted to the 53rd IEEE Conference on Decision and Control, with identical content but a different layout.

Further, the difficult boundary conditions become homogeneous (zero) in the log-transformed variable. The result is an eigenvalue problem that can readily be solved using e.g. the Galerkin method, and sometimes even analytically.

The method requires the cost to be quadratic in the control, and for first order systems there are basically no other significant limitations on how to tune the cost function. However, for the method to work for higher order systems the cost of control can no longer be chosen freely but has to have an inverse relation to the disturbance covariance. This can be a severe limitation and also affects how to find a control policy when the disturbance is no longer white but of low pass character [14].

Here we introduce a transformation of the gradient of the cost function instead of the actual cost, which gives additional degrees of freedom that allow this problem to be circumvented. The transformation results in the same eigenvalue problem as before, a set of PDE:s and a point-wise quadratic equation. For a given solution to the eigenvalue problem the PDE:s are linear, which indicates that the problem can be solved efficiently by an iterative scheme.

Using an LQG problem with state constraints we show how a close approximation to an optimal control law can be found by solving a small nonlinear programming (NLP).

2 Problem formulation

Consider a stochastic dynamic system which is affine in the control inputs:

$$\dot{x} = f(x) + G(x)(u + w), \tag{1}$$

where $x \in \Omega \subset \mathbb{R}^n$ is the state of the system, $u \in \mathbb{R}^m$ is the control signal, $f \in (\Omega \to \mathbb{R}^n)$ and $G \in (\Omega \to \mathbb{R}^{n \times m})$ are functions that describe the system dynamics, and $w \in \mathbb{R}^m$ is a Gaussian white noise having a covariance $W \in (\Omega \to \mathbb{R}^{m \times m})$ that may depend on the state.

We aim to determine a feedback control policy that minimizes the expected cost λ, which can be an arbitrary function of x, though quadratic in u:

$$\lambda = \mathrm{E}\left\{ l(x) + u^T R(x) u \right\}, \tag{2}$$

where $l \in (\Omega \to \mathbb{R})$ is a penalty on the state that the system is in, and $R \in (\Omega \to \mathbb{R}^{m \times m})$ defines the cost of the control.

We assume that the system and the cost are such that an optimal control law exists. Additionally, we require both W and R to be positive definite and bounded everywhere on Ω, but otherwise impose no restrictions on them. Contrary to the assumptions in previous work [9, 10, 14] and the work of Kappen [7] and Broek et al. [12] they are no longer required to relate to the inverse of eachother. As formulated, the control u and the noise w enters the state equation via the same matrix G. However, the problem can easily be reformulated such that the control and noise enter via different matrices as long as they have the same column space [14]. Models where some noise components cannot be directly counteracted by control action are excluded though.

The Hamilton-Jacobi-Bellman equation for this optimization problem can be written as [3]

$$-\frac{\partial V}{\partial t} = \min_u \{ l + (\nabla V)(f + Gu) + u^T R u + \frac{1}{2} \mathrm{tr}\big[(\nabla^T \nabla V) G W G^T\big] \}, \tag{3}$$

where we use a notation where all vectors, such as x, u, w and f, are column vectors. The only exception is ∇, which is a row vector of partial derivative operators $\left(\frac{\partial}{\partial x_1}, \frac{\partial}{\partial x_2}, \ldots\right)$, implying that gradients such as ∇V are row vectors.

Assuming that the process is ergodic, the expected cost per unit time will eventually become the same everywhere in Ω. The stationary solution to (3) minimizing

$$\lambda = -\frac{\partial V}{\partial t} \tag{4}$$

then gives the optimal feedback

$$u = -\frac{1}{2} R^{-1} G^T \nabla^T V, \tag{5}$$

where ∇V is the unknown to be determined. Inserting this expression into (3) results in a nonlinear partial differential equation in V, which is generally difficult to solve. In particular, the state constraint $x \in \Omega$ makes it troublesome since it translates into an infinite cost ($V \to \infty$) on the boundary ($x \in \partial\Omega$).

3 Variable transformation

Let K be a real-valued, positive definite $n \times n$ matrix that is a function of x, and let Z be a real-valued, non-negative scalar function of x. Now, let

$$\nabla V := \frac{-2}{Z} (\nabla Z) K. \tag{6}$$

Since this defines the gradient of V, rather than V itself, we need to make sure that V is well-defined. Because our domain is contractible, though, Poincaré's lemma guarantees that V exists if its Hessian is symmetric [13]. We obtain the Hessian of V by differentiating (6):

$$\nabla^T\nabla V = \frac{-2}{Z}\left((I_n \otimes \nabla Z)\nabla \text{vec}(K) + (\nabla^T\nabla Z)K - \frac{2}{Z}P\right), \tag{7}$$

where

$$P := (\nabla Z)^T(\nabla Z)K, \tag{8}$$

$\otimes$ denotes the Kronecker product, I_n is the $n \times n$ identity matrix and vec(K) is a column vector of the elements in K, taken in column first order:

$$\text{vec}(K) = \begin{pmatrix} k_{11} & \dots & k_{1n} & k_{21} & \dots & k_{nn} \end{pmatrix}^T. \tag{9}$$

The fact that the Hessian matrix is symmetric can be written formally as

$$\nabla^T\nabla V = (\nabla^T\nabla V)^T, \tag{10}$$

which gives one partial differential equation for each pair of off-diagonal elements, i.e. $n(n-1)/2$ equations.

On the boundary, an unbounded control signal is required to counteract the unbounded noise. This means that both ∇V and V go to infinity. By (6) this translates to a boundary condition

$$Z = 0, \quad x \in \partial\Omega. \tag{11}$$

Since (7) can be multiplied by a scalar, e.g. Z^2, without affecting the symmetry of the matrix, we need not worry about the coefficients becoming infinite as $Z \to 0$. Further, the term $2P/Z^2$ in (7) clearly dominates as $Z \to 0$, and therefore also P must be symmetric on the boundary:

$$P^T = P, \quad x \in \partial\Omega. \tag{12}$$

Inserting (6) into (5) gives

$$u = \frac{1}{Z}R^{-1}G^T(\nabla Z K)^T \tag{13}$$

Now, inserting (6), (7) and (13) into the HJB equation (3), and multiplying with Z, gives

$$\lambda Z = lZ - 2(\nabla Z)Kf - \frac{\rho}{Z} - \text{tr}\big[\big((I_m \otimes \nabla Z)\nabla \text{vec}(K) + (\nabla^T\nabla Z)K\big)\, GWG^T\big], \tag{14}$$

where ρ is a quadratic form in $(\nabla Z)K$:

$$\rho := (\nabla Z)KGR^{-1}G^TK^T(\nabla Z)^T - (\nabla Z)GWG^TK^T(\nabla Z)^T \tag{15}$$

Chosing K such that

$$\rho = 0 \tag{16}$$

makes (14) a linear eigenvalue problem in Z. It is worth noting that the solutions to (12) and (16) do not depend on the magnitude of ∇Z, only its direction. Also worth noting is that (16) is satisfied if the columns of G are linearly independent, and

$$K = GWRG^{\dagger}, \tag{17}$$

where $G^{\dagger}$ is the Moore–Penrose pseudoinverse of G. Although this choice of K will not satisfy (10) in general, it actually does when the system can be decoupled into independent one-dimensional optimal control problems.

On the boundary $\partial\Omega$ the direction of ∇Z is fixed (perpendicular to the boundary) and (16) and (12) then define the boundary conditions on K (independent of the exact shape of Z).

4 Solving the problem

We can now determine a solution to the original optimal control problem by

- solving (14) without the ρ-term. For a fixed K, this is a linear elliptic eigenvalue problem, where the principal eigenfunction is sought.
- solving the system (10) of PDE:s for the symmetry of the Hessian of V. For a fixed Z, these PDE:s are linear in K.
- combining the above with (15) and (16) such that the ρ-term disappears from (14). For a fixed direction of ∇Z, this is a quadratic equation in K for every point in Ω.

The free variables are Z as well as each of the $n \times n$ elements of K. Hence, in each point on Ω we have $n^2 + 1$ degrees of freedom, but we only have $n(n-1)/2 + 2$ equations. This leaves $n(n+1)/2 - 1$ degrees of freedom that we can use in any way we like to facilitate the numerical solution of the problem. A common way to utilize excess degrees of freedom is to introduce an optimization objective. For example, the objective

$$\phi = \int_{\Omega} \sum_i \sum_j \|\nabla K_{ij}\|^2 \mathrm{d}\Omega \tag{18}$$

means that the solution that minimizes ϕ should have components of K that are smooth in x.

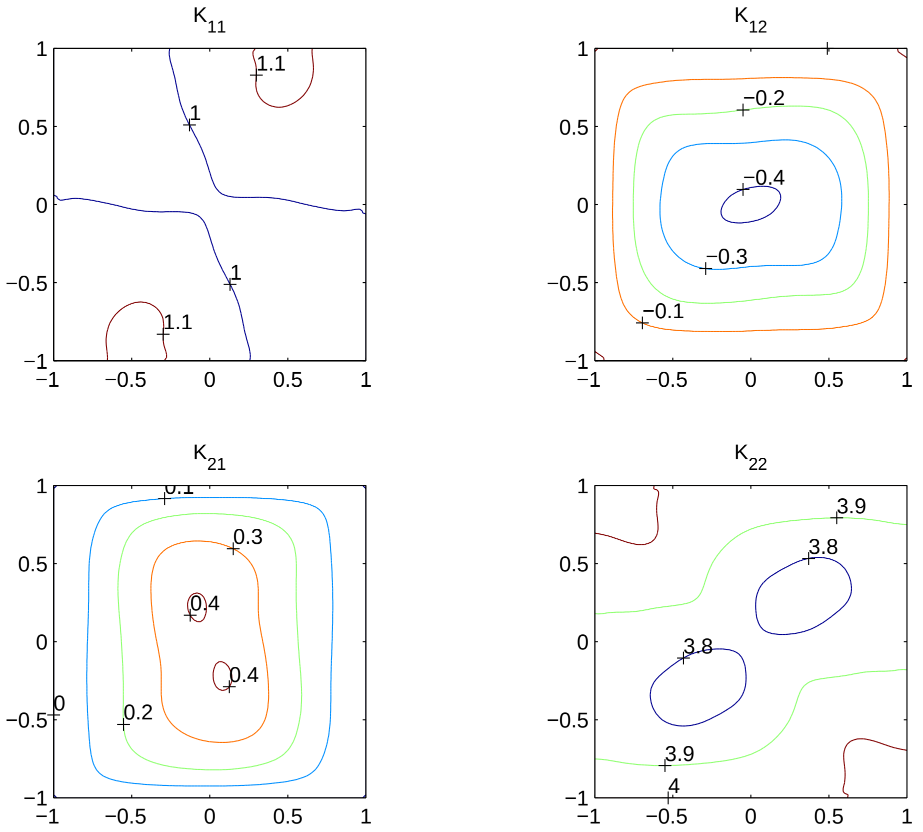

Figure 1: Iso-curves for the elements of the matrix K (one plot for each of the elements in the 2×2 matrix).

The fastest way to solve this problem is probably to iterate between solving for Z and solving for K, but for the example presented in the next section we have simply used a NLP solver to solve for all variables simultaneously.

The nonlinear optimization solver minimizes ϕ at the same time as it solves (16), (14) and (10). As a starting guess for K we can use (17), and as a starting guess for Z we can solve (14) for $\rho = 0$, which is a linear eigenvalue problem given our starting guess for K.

5 Numeric example

As an illustration of the method, we solve a 2-dimensional optimal control problem using a collocation method.

We consider the system given in (1), with $f(x) = Ax$, $l(x) = x^T Qx$, and

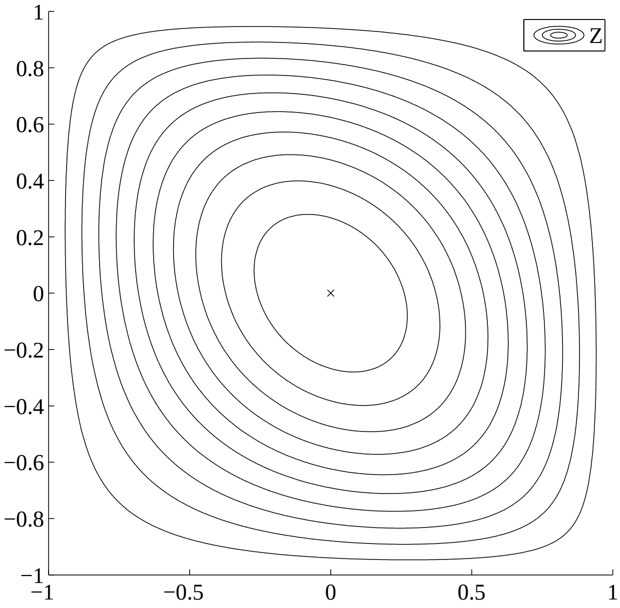

Figure 2: Iso-curves for the solution Z to the eigenvalue problem.

the following numeric parameters:

$$A = \begin{pmatrix} 0 & 1 \\ 0 & 0 \end{pmatrix} \tag{19}$$

$$G = \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix} \tag{20}$$

$$Q = \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix} \tag{21}$$

$$R = \begin{pmatrix} 1 & 0 \\ 0 & 4 \end{pmatrix} \tag{22}$$

$$W = \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix} \tag{23}$$

on the domain

$$-1 \leq x_1 \leq 1 \tag{24}$$

$$-1 \leq x_2 \leq 1. \tag{25}$$

This is a Linear Quadratic Gaussian (LQG) system, but with state constraints. The A-matrix is that of a double integrator, though inputs are

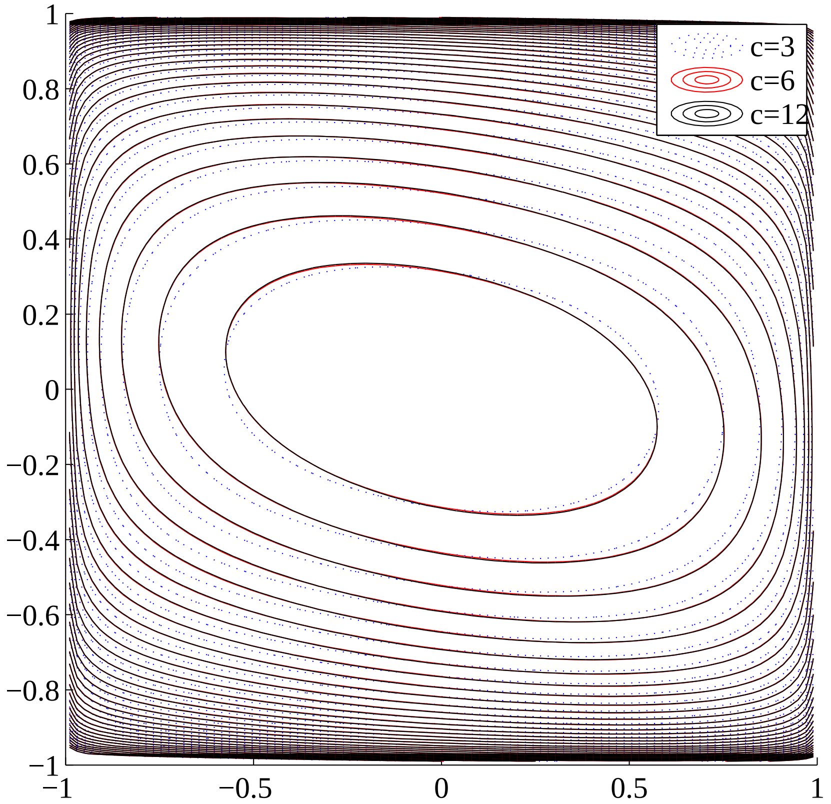

Figure 3: Iso-curves for the solution V to the HJB equation of our original system. The plot does not extend all the way to the boundary. If it did, there would be an infinite number of curves. In an analogue to Hilbert's hotel paradox, each curve goes all the way around, yet the curves are more densely packed along the top and bottom than along the left and right edges.

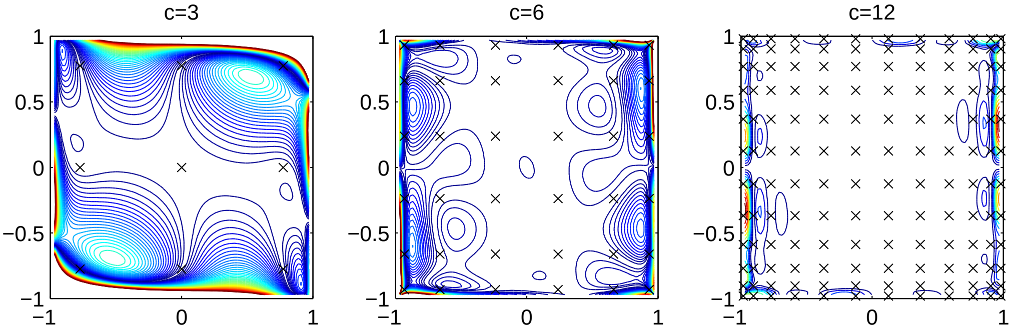

Figure 4: Contour plots of the absolute value of the residual of the HJB equation (3). The distance between curves corresponds to 0.02λ. The collocation points, where the residual is zero by construction, are marked with black crosses. (The residual is unbounded close to the boundary, where several terms in the HJB equation go to infinity, but the expected value of the residual is still bounded, because the probability goes to zero faster than the residual diverges.)

affecting both states here. (The case with only one input is easier, and can be solved using an exact linearization of the HJB equation as in [10].)

The system of equations is solved using collocation [2]. We take the Gauss–Legendre points in each dimension as collocation points, and approximate Z and the elements of K as m-dimensional polynomials of appropriate degree. To set up the collocation problem the TOMLAB/PROPT [6, 11] software was used, and the resulting nonlinear problem was solved using TOMLAB/SNOPT [5]. The resulting K and Z are shown in Figures 1 and 2 respectively.

In order to obtain a starting guess, we initially set the off-diagonal element of A to zero, which results in a separable system of two one-dimensional optimal control problems with known analytic solutions [8].

Note that we do not actually need to compute V if we are only interested in the control policy. The control depends on ∇V, but not on V itself, and we can compute it from Z and K using (13). However, we did compute Z (for plotting purposes) by selecting a reference point and computing line integrals from that point to every other point. Figure 3 shows the result for a varying number of collocation points. In theory, it does not matter which path we choose to integrate along. In practice though, there will always be a residual error in the symmetry equations such that the choice of path has a slight effect on the computed Z. For the plots presented here, we integrated along straight lines to the origin, although it could be argued that integrating along the direction of the gradient would be a better choice.

6 Accuracy and mesh size

When solving partial differential equations numerically, the mesh size, or in this case the number of collocation points in each dimension, always plays an important role. Using a finer mesh, or more points, usually gives a smaller residual error but at the expense of requiring more computational resources.

Optimal control problems often involve a high number of states, and any method that relies on parametrization a function in the state space will need an exponential number of parameters. This is known as "the curse of dimensionality".

The method derived here also suffers from this curse. The number p of free parameters that are solved for is

$$p = (n^2 + 1)c^n \tag{26}$$

where c is the number of collocation points in each dimension. This number depends on how smooth the sought functions are. If Z and the elements of K can be approximated by low-order polynomials on Ω, then c can be small.

Even though p is exponential in n, we can consider relatively large systems if c is small enough. Figure 3 shows solutions for $c = 3$, $c = 6$, and $c = 12$. There is a clearly visible difference between the solutions for 3×3 points (blue dots), and the solution for 6×6 points (red). However, the solution for 6×6 and 12×12 points (black lines) are nearly identical. Although three points per dimension resulted in a slightly suboptimal solution, this low number of points per dimension should make it feasible to solve optimal control problems in relatively high dimension.

The solution for $c = 3$, although slightly different from the other two, looks like it may be acceptable. Letting $c = 3$ and $n = 8$ gives $p = 426465$. A problem in eight states would thus require solving for less than half a million unknowns, which should be feasible on an ordinary personal computer with a good numeric algorithm.

Figure 2 shows the solution Z to the eigenvalue problem. Unlike V in Figure 3, Z is smooth all the way out to the boundary, and is therefore well-approximated by a low-order polynomial.

Figure 4 shows the absolute value of the residual of the HJB equation (3) for the same solutions as in Figure 3. It can clearly be seen how the residual decreases with an increasing number of collocation points.

7 Discussion and future work

The numerical results verified that the method works for this example. In particular, it is very encouraging that the solution using very few collocation points is very similar to the ones with a large number of points. However, more work is needed to determine the circumstances in which this method is applicable.

Another interesting issue is whether the set of PDE:s (10) always has a solution. We know that V exists, because we have chosen our system such that an optimal control policy exists. We also know that Z exists, because it is the principal eigenvalue of an elliptic partial differential equation. This means that, knowing V, we could compute a K to match a given Z as long as $\nabla Z \neq 0$. For $\nabla Z = 0$, equation (6) gives $\nabla V = 0$, so Z will have the same optimum as V if K is positive definite everywhere.

We used a commercially available NLP solver to solve for K and Z at once. This allowed us to quickly test the method, but at the expense of computational effort. The next step is to develop an iterative algorithm, as outlined in Section 4, to address convergence rate and other numerical issues.

8 Conclusion

We have derived and demonstrated a method for solving the HJB equation for a stochastic system with state constraints. Rather than solving for the cost-to-go function V directly, we solved for a matrix function and an eigenfunction that together define the control policy. Although we needed to solve for a larger number of functions, these functions could be well approximated by low order polynomials, such that the number of numeric parameters in the parametrization of the solution was low.

This suggests that the presented method, combined with an efficient numeric algorithm, is a good way to mitigate Bellman's so-called curse of dimensionality, and solve the HJB equation for these kinds of systems in higher dimension than what has previously been feasible.

References

[1] Richard Bellman. *Dynamic Programming.* Princeton University Press, 1957.

[2] J.P. Boyd. *Chebyshev and Fourier Spectral Methods.* Dover books on mathematics. Dover Publications, 2001.

[3] Peter Dorato, Chaouki Abdallalh, and Vito Cerone. *Linear-Quadratic Control, An Introduction.* Prentice-Hall, 1995.

[4] Wendell H. Fleming. Logarithmic transformations and stochastic control. In Wendell H. Fleming and Luis G. Gorostiza, editors, *Advances in Filtering and Optimal Stochastic Control*, volume 42 of *Lecture Notes in Control and Information Sciences*, pages 131–141. Springer Berlin Heidelberg, 1982.

[5] Philip E. Gill, Walter Murray, Michael, and Michael A. Saunders. Snopt: An sqp algorithm for large-scale constrained optimization. *SIAM Journal on Optimization*, 12:979–1006, 1997.

[6] Göran Anders Holmström, Kenneth and Marcus Edvall. User's guide for tomlab 7. Technical report, Tomlab Optimization.

[7] H J Kappen. Path integrals and symmetry breaking for optimal control theory. *Journal of Statistical Mechanics: Theory and Experiment*, 2005(11):P11011, 2005.

[8] Per Rutquist, Claes Breitholtz, and Torsten Wik. An eigenvalue approach to infinite-horizon optimal control. In *Proc. 16th IFAC World Congress*, Prague, Czech Republic, jul 2005.

[9] Per Rutquist, Claes Breitholtz, and Torsten Wik. On the infinite-time solution to state-constrained stochastic optimal control problems. Technical Report R006/2005, Chalmers University of Technology, Göteborg, Sweden, feb 2005.

[10] Per Rutquist, Claes Breitholtz, and Torsten Wik. On the infinite time solution to state-constrained stochastic optimal control problems. *Automatica*, 44(7):1800 – 1805, 2008.

[11] Per Rutquist and Marcus Edvall. Propt - matlab optimal control software. Technical report, Tomlab Optimizatoin.

[12] Bart van den Broek, Wim Wiegerinck, and Bert Kappen. Stochastic optimal control of state constrained systems. *International Journal of Control*, 84(3):597–615, 2011.

[13] S.H. Weintraub. *Differential Forms: Theory and Practice.* Elsevier Science, 2014.

[14] Torsten Wik, Per Rutqvist, and Claes Breitholtz. State-Constrained Control Based on Linearization of the Hamilton-Jacobi-Bellman Equation. In *49TH IEEE conference on decision and control (CDC)*, pages 5192–5197. IEEE, 2010. 49th IEEE Conference on Decision and Control (CDC), Atlanta, GA, DEC 15-17, 2010.