An Inequality Constrained Nonlinear Kalman-Bucy Smoother by Interior Point Likelihood Maximization ⋆

Bradley M. Bell [a], James V. Burke [b], Gianluigi Pillonetto [c]

[a] *Applied Physics Laboratory, University of Washington, Seattle U.S.A.*

[b] *Department of Mathematics, University of Washington, Seattle U.S.A.*

[c] *Department of Information Engineering, University of Padova, Padova Italy*

Abstract

Kalman-Bucy smoothers are often used to estimate the state variables as a function of time in a system with stochastic dynamics and measurement noise. This is accomplished using an algorithm for which the number of numerical operations grows linearly with the number of time points. All of the randomness in the model is assumed to be Gaussian. Including other available information, for example a bound on one of the state variables, is non trivial because it does not fit into the standard Kalman-Bucy smoother algorithm. In this paper we present an interior point method that maximizes the likelihood with respect to the sequence of state vectors satisfying inequality constraints. The method obtains the same decomposition that is normally obtained for the unconstrained Kalman-Bucy smoother, hence the resulting number of operations grows linearly with the number of time points. We present two algorithms, the first is for the affine case and the second is for the nonlinear case. Neither algorithm requires the optimization to start at a feasible sequence of state vector values. Both the unconstrained affine and unconstrained nonlinear Kalman-Bucy smoother are special cases of the class of problems that can be handled by these algorithms.

Key words: Kalman filter; inequality constrained systems; state estimation; function estimation; interior point applications.

1 Introduction

The discrete Kalman-Bucy model for the dynamics and measurement is

$$x_k = g_k(x_{k-1}) + w_k \quad , \quad z_k = h_k(x_k) + v_k \tag{1}$$

where for time index $k = 1, \ldots, N$, $x_k \in \mathbf{R}^n$ is the unknown state vector, $g_k : \mathbf{R}^n \to \mathbf{R}^n$ is the known model for the mean dynamics, $w_k \in \mathbf{R}^n$ is the unknown noise in the dynamics, $z_k \in \mathbf{R}^m$ is the known measurement vector, $h_k : \mathbf{R}^n \to \mathbf{R}^m$ is the known model for the measurement mean, and $v_k \in \mathbf{R}^m$ is the unknown noise in the measurements. The noise vectors are all mutually independent and

$$w_k \sim \mathbf{N}(0, Q_k) \quad , \quad v_k \sim \mathbf{N}(0, R_k) \tag{2}$$

where $Q_k \in \mathbf{R}^{n \times n}$ and $R_k \in \mathbf{R}^{m \times m}$ are the known covariance matrices, hence symmetric and positive definite. We simplify the notation by using $g_1(x_0)$ for our initial state estimate, where g_1 is a constant function, and using Q_1 for the covariance of this initial estimate. We also simplify notation by fixing the number of measurements at each time index to m. The algorithms allow some or all of these measurement to be missing at any particular time index so this is not a restriction.

If h_k and g_k are affine (linear plus a constant), the maximum likelihood and minimum variance estimate for the state sequence $\{x_k : k = 1, \ldots, N\}$ are identical and can be obtained using the affine Kalman-Bucy smoother; e.g., [1, Fig 1] or [23]. A decomposition technique enables these algorithms to solve nN linear equations in nN variables in $O(n^3 N)$ operations. If h_k and g_k are nonlinear, the iterated Kalman-Bucy smoother can be used to determine the maximum likelihood estimate for the

⋆ This research was supported in part by NIH Grant P41 EB 001975, NSF Grant DMS-0505712, and the PRIN Project *New Methods and Algorithms for Identification and Adaptive Control of Technological Systems.*

Email addresses: bradbell@washington.edu (Bradley M. Bell), burke@math.washington.edu (James V. Burke), giapi@dei.unipd.it (Gianluigi Pillonetto).

state sequence. This is a Gauss-Newton method where each iteration of the method uses the affine Kalman-Bucy smoother (e.g. [1, Fig 2]).

In many cases, the model for x_k in (1) is only an approximation that can be improved by including other information or considerations about the state vector in the estimation procedure (for further discussion see [24, Section 2.2]). The equality constrained solution where the model and constraint functions are affine is well known; e.g. [26]. Nonlinear models with nonlinear equality constraints are often approximated by augmenting the measurement vector; e.g., [11,30]. There is much recent work addressing inequality constraints using moving horizon estimation; e.g., [21,22,28,34]. The moving horizon is used to reduce the optimization problem in order to make it tractable; see the discussion below [8, equation (9)]. The interior point method presented below obtains the same decomposition of the linear equations as for the unconstrained affine Kalman-Bucy smoother (see Theorem 3). This makes it feasible to simultaneously optimize with respect to the state value at all time indices even for very large values of N. We use $f_k : \mathbf{R}^n \to \mathbf{R}^\ell$ to denote known functions and model the constraints by

$$f_k(x_k) \leq 0 \quad \text{for} \quad k = 1, \ldots, N \tag{3}$$

Active set methods are a popular approach to inequality constrained optimization problems. However, these methods have a worst case complexity that is exponential in number of constraints ℓN [9, equation (25)] which is a significant issue for large N. The problem with active set methods is that even state of the art implementations cannot guarantee that the optimal active set has been identified until successful termination [4,7,12,17,32]. Interior point methods are, on the other hand, fundamentally different. Indeed, they may be thought of as the opposite of active set methods since no constraint is allowed to become active as the iterations proceed. Our approach is motivated by the great practical success of interior point methods which have revolutionized the theory and practice of numerical optimization over the past 25 years. This approach to the inequality constrained nonlinear Kalman-Bucy smoother is in the same spirit as the approach taken to certain optimal control problems in [33].

In Section 2, we introduce a nonlinear Kalman-Bucy smoother problem with inequality constraints. This problem is approximated by a Quadratic Program (QP) that corresponds to the affine Kalman-Bucy smoother with affine inequality constraints (Section 3). The approximating QP is solved using an interior point method that maximizes the likelihood with respect to the entire state sequence $\{x_k : k = 1, \ldots, N\}$. The interior point method requires only $O(n^3 N)$ operations for each interior point iteration (Sections 4 and 5). As an introduction to the constrained nonlinear algorithm, an unconstrained version is presented (Section 6). As in sequential quadratic programming methods for nonlinear programs [19], the QP is iterated to obtain a method for solving the nonlinear constrained Kalman-Bucy smoother problem (Section 7). A global convergence proof for the corresponding algorithm is included in Section 7. A constrained smoothing spline example is presented as an application of the affine Kalman-Bucy smoother with affine inequality constraints (Section 8). A ship tracking example is presented as an application of the nonlinear Kalman-Bucy smoother with nonlinear inequality constraints (Section 9).

2 The Nonlinear Problem

Let $\{x_k\}$ denote the entire state sequence $\{x_k : k = 1, \ldots, N\}$ with a similar notation for all other indexed quantities in equations (1)-(3). The probability density corresponding to a multivariate normal distribution is given by [14, equation 2.80]. In particular, for the Kalman-Bucy model (1)-(2), the negative log of the probability density for the measurement sequence $\{z_k\}$ given the state sequence $\{x_k\}$ is

$$\begin{aligned} -\log \mathbf{p}(\{z_k\}|\{x_k\}) = \frac{1}{2}\sum_{k=1}^{N} &\log\det(2\pi R_k) \\ &+ [z_k - h_k(x_k)]^{\mathrm{T}} R_k^{-1} [z_k - h_k(x_k)] \end{aligned}$$

Prior to making the measurements, the negative log of the probability density for the state sequence is

$$\begin{aligned} -\log \mathbf{p}(\{x_k\}) = \frac{1}{2}\sum_{k=1}^{N} &\log\det(2\pi Q_k) \\ &+ [x_k - g_k(x_{k-1})]^{\mathrm{T}} Q_k^{-1} [x_k - g_k(x_{k-1})] \end{aligned}$$

Define $S_k : \mathbf{R}^n \times \mathbf{R}^n \to \mathbf{R}$ to be the residual sum of squares corresponding to time index k,

$$\begin{aligned} S_k(x_k, x_{k-1}) \;=\; & \tfrac{1}{2}[z_k - h_k(x_k)]^{\mathrm{T}} R_k^{-1} [z_k - h_k(x_k)] \\ & + \tfrac{1}{2}[x_k - g_k(x_{k-1})]^{\mathrm{T}} Q_k^{-1} [x_k - g_k(x_{k-1})] \end{aligned} \tag{4}$$

Define $S : \{\mathbf{R}^n\} \to \mathbf{R}$ to be the total sum of squares

$$S(\{x_k\}) = \sum_{k=1}^{N} S_k(x_k, x_{k-1}).$$

Given our model formulation, $S(\{x_k\})$ can be computed for any input sequence $\{x_k\}$; furthermore,

$$\begin{aligned} \mathbf{p}(\{z_k\},\ \{x_k\}) &= \mathbf{p}(\{z_k\}|\{x_k\})\ \mathbf{p}(\{x_k\}) \\ \partial_{x(j)} \log \mathbf{p}(\{z_k\},\ \{x_k\}) &= -\partial_{x(j)} S(\{x_k\}) \end{aligned}$$

(Here and below we use $w(p)$ to denote w_p when it is a subscript.) It follows that $S(\{x_k\})$ is equal to a constant plus the negative log of the joint probability density of $\{z_k\}$ and $\{x_k\}$. Thus, minimizing $S(\{x_k\})$ subject to a set of constraints is equivalent maximizing the joint density subject to the same set of constraints. Our inequality constrained nonlinear Kalman-Bucy smoother problem is

$$\begin{array}{ll} \text{minimize} & S(\{x_k\}) \quad \text{w.r.t. } \{x_k\} \\ \text{subject to} & f_k(x_k) \leq 0 \ \ (k = 1, \ldots, N). \end{array} \tag{5}$$

Remark 1 *There is an inherent contradiction in the model formulation (1)-(3). The constraints (3) for the state sequence $\{x_k\}$ conflict with the model for the noise sequence $\{w_k\}$, and so the model for noise cannot be completely accurate. In the case of linear equality constraints, there are computable consistent statistical models; e.g., [15]. In the nonlinear inequality constrained, for computational as well as modeling simplicity, it is useful to think of the components of $\{w_k\}$ as truncated Gaussian distributions so that the lemma makes sense.*

3 The Quadratic Programming Sub-problem

Our grand strategy for solving problem (5) is to build an algorithm that is reminiscent of the iterative Kalman-Bucy filter for the unconstrained problem. We do this by iteratively solving QP approximations that are built around the current estimate of a solution. These QP approximations correspond to local Gauss-Newton approximations to problem (5). We propose to solve these QP subproblems using an interior point approach that exploits their structure in much the same way as the Kalman-Bucy filter. We now turn to the development of our Gauss-Newton approximation.

It follows from the definition of S_k in equation (4) that

$$\partial_{x(j)} S_k(x_k, x_{k-1}) = \begin{cases} \left\{ \begin{array}{c} [x_k - g_k(x_{k-1})]^{\mathrm{T}} Q_k^{-1} \\ -[z_k - h_k(x_k)]^{\mathrm{T}} R_k^{-1} h_k^{(1)}(x_k) \end{array} \right\} & \text{, if } j = k \\ -[x_k - g_k(x_{k-1})]^{\mathrm{T}} Q_k^{-1} g_k^{(1)}(x_{k-1}) & \text{, if } j = k-1 \\ 0 & \text{, otherwise} \end{cases}$$

Hence, except for the case $j = N$,

$$\begin{aligned} \partial_{x(j)} S(\{x_k\}) &= \sum_{k=1}^{N} \partial_{x(j)} S_k(x_k, x_{k-1}) \\ &= \partial_{x(j)} S_j(x_j, x_{j-1}) + \partial_{x(j)} S_{j+1}(x_{j+1}, x_j) \\ &= [x_j - g_j(x_{j-1})]^{\mathrm{T}} Q_j^{-1} \\ &\quad - [z_j - h_j(x_j)]^{\mathrm{T}} R_j^{-1} h_j^{(1)}(x_j) \\ &\quad - [x_{j+1} - g_{j+1}(x_j)]^{\mathrm{T}} Q_{j+1}^{-1} g_{j+1}^{(1)}(x_j) \end{aligned} \tag{6}$$

For the case where $j = N$, there is no term $S_{j+1}(x_{j+1}, x_j)$ in the summation. Thus, we obtain the following lemma.

Lemma 2 *Define $Q_{N+1} \in \mathbf{R}^{n \times n}$ to be the identity matrix and $g_{N+1} \equiv 0$. Then the partial derivative of $S(\{x_k\})$ with respect to $x(j)$ is given by equation (6) for $j = 1, \ldots, N$.*

We now build an affine approximation to the model (1)-(3) that is first-order accurate for a state sequence $\{y_k\}$ near a fixed state sequence $\{x_k\}$. Define the affine approximations $\tilde{f}_k$, $\tilde{g}_k$, and $\tilde{h}_k$ by

$$\begin{aligned} \tilde{f}_k(x_k; y_k) &= f_k(x_k) + f_k^{(1)}(x_k)(y_k - x_k) \\ \tilde{g}_k(x_k; y_k) &= g_k(x_k) + g_k^{(1)}(x_k)(y_k - x_k) \\ \tilde{h}_k(x_k; y_k) &= h_k(x_k) + h_k^{(1)}(x_k)(y_k - x_k) \end{aligned}$$

and, correspondingly, define $\tilde{S}(\{x_k\}; \{y_k\})$ to be the residual sum of squares function associated with these affine approximations:

$$\begin{aligned} &\tilde{S}_k(x_k, x_{k-1}; y_k, y_{k-1}) = \\ &\quad (1/2)[y_k - \tilde{g}_k(x_{k-1}; y_{k-1})]^{\mathrm{T}} Q_k^{-1} [y_k - \tilde{g}_k(x_{k-1}; y_{k-1})] \\ &\quad + (1/2)[z_k - \tilde{h}_k(x_k; y_k)]^{\mathrm{T}} R_k^{-1} [z_k - \tilde{h}_k(x_k; y_k)] \end{aligned}$$

$$\tilde{S}(\{x_k\}; \{y_k\}) = \sum_{k=1}^{N} \tilde{S}_k(x_k, x_{k-1}; y_k, y_{k-1}) \tag{7}$$

It follows from these definitions that

$$S(\{x_k\}) = \tilde{S}(\{x_k\}; \{x_k\}) \tag{8}$$

Next fix the state sequence $\{x_k\}$ and use $\tilde{f}_k$, $\tilde{g}_k$, and $\tilde{h}_k$ in place of f_k, g_k, and h_k to define a Kalman-Bucy model (1)-(3) for the state sequence $\{y_k\}$. Applying the equation (6) to the affine approximation yields

$$\begin{aligned} \partial_{y(j)} \tilde{S}(\{x_k\}; \{y_k\}) &= [y_j - \tilde{g}_j(x_{j-1}; y_{j-1})]^{\mathrm{T}} Q_j^{-1} \\ &\quad - [z_j - \tilde{h}_j(x_j; y_j)]^{\mathrm{T}} R_j^{-1} h_j^{(1)}(x_j) \\ &\quad - [y_{j+1} - \tilde{g}_{j+1}(x_j; y_j)]^{\mathrm{T}} Q_{j+1}^{-1} g_{j+1}^{(1)}(x_j) \end{aligned} \tag{9}$$

$$\partial_{x(j)} S(\{x_k\}) = \partial_{y(j)} \tilde{S}(\{x_k\}; \{y_k\}) \Big|_{\{y_k\} = \{x_k\}} \tag{10}$$

We conclude from equations (8) and (10) that $\tilde{S}(\{x_k\}; \{y_k\})$, as a function of $\{y_k\}$, is a first-order accurate approximation to $S(\{x_k\})$ near $\{x_k\}$. In addition, $\tilde{S}(\{x_k\}; \{y_k\})$ is smooth with respect to $\{x_k\}$ (provided that $g_k(x_{k-1})$ and $h_k(x_k)$ are smooth).

Our approach to solving the nonlinear problem (5) is based on iteratively solving QP subproblems of the form:

$$\begin{array}{l} \text{minimize} \quad \tilde{S}(\{x_k\};\{y_k\}) \ \text{ w.r.t. } \{y_k\} \\ \text{subject to } \ \tilde{f}_k(x_k;y_k) \le 0 \ \ (k=1,\ldots,N) \end{array} \tag{11}$$

This approach is similar in spirit to sequential quadratic programming (SQP) methods for constrained nonlinear programs [19].

Let us analyze the structure of these subproblems a bit further. It follows from equation (9) that the second partials of $\tilde{S}(\{x_k\},\{y_k\})$ with respect to $\{y_k\}$ are zero except for the terms

$$\begin{aligned} &\partial_{y(j-1)}\partial_{y(j)}\tilde{S}(\{x_k\};\{y_k\}) = -g_j^{(1)}(x_{j-1})^{\mathrm{T}}Q_j^{-1} \\ &\partial_{y(j)}\partial_{y(j)}\tilde{S}(\{x_k\};\{y_k\}) = Q_j^{-1} \\ &\quad +h_j^{(1)}(x_j)^{\mathrm{T}}R_j^{-1}h_j^{(1)}(x_j) + g_{j+1}^{(1)}(x_j)^{\mathrm{T}}Q_{j+1}^{-1}g_{j+1}^{(1)}(x_j) \\ &\partial_{y(j+1)}\partial_{y(j)}\tilde{S}(\{x_k\};\{y_k\}) = -Q_{j+1}^{-1}g_{j+1}^{(1)}(x_j) \end{aligned}$$

for $j = 1,\ldots,N$. Also note that the Hessian of $\tilde{S}(\{x_k\},\{y_k\})$ with respect to $\{y_k\}$ is constant with respect to $\{y_k\}$, that is, $\tilde{S}(\{x_k\},\{y_k\})$ is quadratic in $\{y_k\}$. Define $A_k \in \mathbf{R}^{n\times n}$, $C_k \in \mathbf{R}^{n\times n}$, and $C \in \mathbf{R}^{nN\times nN}$ by

$$\begin{aligned} A_k &= -Q_k^{-1}g_k^{(1)}(x_{k-1}) \\ C_k &= \left\{ \begin{array}{l} Q_k^{-1} + h_k^{(1)}(x_k)^{\mathrm{T}}R_k^{-1}h_k^{(1)}(x_k) \\ \quad +g_{k+1}^{(1)}(x_k)^{\mathrm{T}}Q_{k+1}^{-1}g_{k+1}^{(1)}(x_k) \end{array} \right\} \\ C &= \begin{pmatrix} C_1 & A_2^{\mathrm{T}} & 0 & \\ A_2 & C_2 & A_3^{\mathrm{T}} & 0 \\ 0 & \ddots & \ddots & \ddots \\ & 0 & A_N & C_N \end{pmatrix} \end{aligned} \tag{12}$$

In Theorem 3 we will see that equations involving the matrix C can be inverted in a stable and efficient fashion. This matrix is the Hessian of $\tilde{S}(\{x_k\},\{y_k\})$ with respect to $\{y_k\}$ where the dependence on $\{x_k\}$ is not explicitly expressed. We now use this simplified notation to reformulate the QP subproblem (11). Define the vector $a_k \in \mathbf{R}^n$ by

$$\begin{aligned} a_k &= \partial_{y(k)}\tilde{S}(\{x_j\};\{y_j\})\Big|_{\{y_j\}=\{x_j\}}^{\mathrm{T}} \\ &= Q_k^{-1}[x_k - g_k(x_{k-1})] - h_k^{(1)}(x_k)^{\mathrm{T}}R_k^{-1}[z_k - h_k(x_k)] \\ &\quad -g_{k+1}^{(1)}(x_k)^{\mathrm{T}}Q_{k+1}^{-1}[x_{k+1} - g_{k+1}(x_k)] \end{aligned}$$

and define the vector $a \in \mathbf{R}^{nN}$, the vector $b \in \mathbf{R}^{\ell N}$, and the matrix $B \in \mathbf{R}^{\ell N\times nN}$ by

$$\begin{aligned} a &= \begin{pmatrix} a_1 \\ \vdots \\ a_N \end{pmatrix} \quad , \quad b = \begin{pmatrix} f_1(x_1) - f_1^{(1)}(x_1)x_1 \\ \vdots \\ f_N(x_N) - f_N^{(1)}(x_N)x_N \end{pmatrix} \\ B &= \begin{pmatrix} f_1^{(1)}(x_1) & 0 & \\ 0 & \ddots & 0 \\ & 0 & f_N^{(1)}(x_N) \end{pmatrix} \end{aligned} \tag{13}$$

The vector a is a representation for the gradient of $\tilde{S}(\{x_k\},\{y_k\})$ with respect to $\{y_k\}$ at $\{y_k\} = \{x_k\}$. The affine approximation to the constraints (3) is given by $b + By \le 0$ where $y \in \mathbf{R}^{nN}$ is the column vector representing $\{y_k\}$. With this notation, the QP subproblem (11) becomes

$$\begin{array}{l} \text{minimize} \quad \frac{1}{2}y^{\mathrm{T}}Cy + d^{\mathrm{T}}y \ \text{ w.r.t. } y \in \mathbf{R}^{nN} \\ \text{subject to } \ b + By \le 0 \end{array} \tag{14}$$

where $d = a - Cx$ and $x \in \mathbf{R}^{nN}$ is the column vector representing $\{x_k\}$. This reformulation is obtained from

$$\begin{aligned} &\tilde{S}(\{x_k\};\{y_k\}) = \\ &\quad (1/2)(y-x)^{\mathrm{T}}C(y-x) + a^{\mathrm{T}}(y-x) + S(\{x_k\}) \end{aligned}$$

by removing the terms that are constant with respect to y. Since the matrix C is positive definite (see Theorem 3), this is a strictly convex QP, and so a unique solution exists whenever the problem is feasible.

4 Solving the QP Sub-problem

The QP subproblem (14) is solved using an interior point approach [16,19]. Interior point methods apply a damped Newton's method to a relaxation of the Karush-Kuhn-Tucker (KKT) conditions. The relaxed optimality conditions are themselves optimality conditions for an associated relaxed optimization problem. The relaxed subproblem uses a log barrier to maintain strict feasibility (see [13, Chapter 3] for a discussion log barrier methods):

$$\begin{array}{l} \text{minimize } (1/2)y^{\mathrm{T}}Cy + d^{\mathrm{T}}y - \mu\sum_{i=1}^{\ell N}\log(s_i) \ \text{ w.r.t} \\ \quad (y,s) \in \mathbf{R}^{nN}\times\mathbf{R}_+^{\ell N} \ \text{ subject to } s + b + By = 0 \end{array} \tag{15}$$

where $s_i \in \mathbf{R}_+$ means that $s_i \ge 0$, and μ is the relaxation parameter. One can show [16] that if the original problem is strictly feasible, i.e., there exists y such that $b + By < 0$, then a unique solution to problem (15) exists for all $\mu > 0$. The expression $-\mu\sum_{i=1}^{\ell N}\log(s_i)$ in the objective

is called a log-barrier term. This term assures that at the solution $(y(\mu), s(\mu))$ to problem (15) one has $0 < s(\mu)$.

The first-order necessary conditions for optimality in problem (15), or equivalently, the KKT conditions, can be stated in terms of its Lagrangian function

$$L_\mu(y,s,u) = \frac{1}{2}y^{\mathrm{T}}Cy + d^{\mathrm{T}}y - \mu\sum_{i=1}^{\ell N}\log(s_i) + u^{\mathrm{T}}(s+b+By)$$

where $u \in \mathbf{R}^{\ell N}_+$ is the vector of Lagrange multipliers corresponding to the constraint $s + b + By = 0$. These optimality conditions take the form

$$\begin{aligned} 0 &= \nabla_u L_\mu(y,s,u) = s + b + By \\ 0 &= \nabla_y L_\mu(y,s,u) = Cy + B^{\mathrm{T}}u + d \\ 0 &= \nabla_s L_\mu(y,s,u) = u - \mu D(s)^{-1}e \end{aligned} \tag{16}$$

for $y \in \mathbf{R}^{nN}$ and $s, u \in \mathbf{R}^{\ell N}_+$, where, for any vector $w \in \mathbf{R}^q$, $D(w) \in \mathbf{R}^{q\times q}$ denotes the diagonal matrix with w along its diagonal (i.e., $D(w)_{i,j}$ is w_i if $i = j$ and zero otherwise), and $e \in \mathbf{R}^{\ell N}$ is the vector having all components equal to one. We now transform these equations into a form that is more convenient for the application of Newton's method. Define

$$F_\mu : \mathbf{R}^{\ell N+nN+\ell N} \to \mathbf{R}^{\ell N+nN+\ell N}$$
$$F_\mu(s,y,u) = \begin{pmatrix} s+b+By \\ Cy + B^{\mathrm{T}}u + d \\ D(s)D(u)e - \mu e \end{pmatrix}$$

Since problem (15) is a convex program with only polyhedral constraints, the point (s, y) solves problem (15) if and only if there is a multiplier vector $u \in \mathbf{R}^{\ell N}_+$ such that (s, y, u) solves the KKT conditions (16), or equivalently, $F_\mu(s, y, u) = 0$ [25]. Our solution method for the QP subproblems (14) uses a predictor-corrector Newton's method to follow the central path

$$\mathcal{C} = \left\{ [s(\mu), y(\mu), u(\mu)] \;\middle|\; \begin{array}{c} 0 < \mu,\ 0 < s(\mu),\ 0 < u(\mu) \\ F_\mu[s(\mu), y(\mu), u(\mu)] = 0 \end{array} \right\}$$

as μ decreases to zero. This is the basic idea of the interior point approach, however, the implementation details that yield an efficient and numerically stable algorithm can be quite delicate. A few of these details are discussed in the remainder of this section.

Let $\mu > 0$ and $(s, y, u) \in \mathbf{R}^{\ell N+nN+\ell N}$ with $0 < s$ and $0 < u$. The Newton step at (s, y, u) for the function F_μ is obtained by solving the equation

$$F^{(1)}_\mu(s,y,u)(\Delta s,\ \Delta y,\ \Delta u)^{\mathrm{T}} = -F_\mu(s,y,u)$$

The derivative of F_μ is given by

$$F^{(1)}_\mu(s,y,u) = \begin{pmatrix} I_{\ell N} & B & 0 \\ 0 & C & B^{\mathrm{T}} \\ D(u) & 0 & D(s) \end{pmatrix}$$

where $I_{\ell N}$ is the $\ell N \times \ell N$ identity matrix. Hence, the Newton iteration becomes

$$\begin{pmatrix} I_{\ell N} & B & 0 \\ 0 & C & B^{\mathrm{T}} \\ D(u) & 0 & D(s) \end{pmatrix} \begin{pmatrix} \Delta s \\ \Delta y \\ \Delta u \end{pmatrix} = \begin{pmatrix} -s-b-By \\ -Cy - B^{\mathrm{T}}u - d \\ \mu e - D(s)D(u)e \end{pmatrix} \tag{17}$$

The first row of this equation gives Δs in terms of Δy:

$$\Delta s = -s - b - B(y + \Delta y) \tag{18}$$

Replacing the bottom row of equation (17) by the bottom row minus $D(u)$ times the top row, we obtain the reduced set of equations

$$\begin{pmatrix} C & B^{\mathrm{T}} \\ -D(u)B & D(s) \end{pmatrix} \begin{pmatrix} \Delta y \\ \Delta u \end{pmatrix} = \begin{pmatrix} -Cy - B^{\mathrm{T}}u - d \\ \mu e + D(u)(b + By) \end{pmatrix}$$

Replacing the top row by C^{-1} times the top row

$$\begin{aligned} \Delta y \quad &+ C^{-1}B^{\mathrm{T}}\Delta u &&= -y - C^{-1}(B^{\mathrm{T}}u + d) \\ -D(u)B\Delta y \ &+ \quad D(s)\Delta u &&= \mu e + D(u)(b + By) \end{aligned} \tag{19}$$

The first row of this equation gives Δy in terms of Δu:

$$\Delta y = -y - C^{-1}[d + B^{\mathrm{T}}(u + \Delta u)] \tag{20}$$

Replacing the bottom row of equation (19) by $D(u)^{-1}$ times the bottom row plus B times the top row, we obtain the reduced equation

$$\begin{aligned} &[BC^{-1}B^{\mathrm{T}} + D(s/u)]\Delta u = \\ &\quad [\mu(e/u) + b - BC^{-1}(B^{\mathrm{T}}u + d)] \end{aligned} \tag{21}$$

where $(s/u) \in \mathbf{R}^{\ell N}_+$ is the vector defined by $(s/u)_i = s_i/u_i$. Having Δu, the vector Δy can be computed using equation (20), and then Δs can be computed using equation (18). Following this approach, a Newton step can be obtained by successively solving the three equations,

$$\begin{aligned} Cz &= B^{\mathrm{T}}u + d \\ \left[BC^{-1}B^{\mathrm{T}} + D(s/u)\right]\Delta u &= [\mu(e/u) + b - Bz] \\ Cw &= d + B^{\mathrm{T}}(u + \Delta u) \end{aligned}$$

and setting $\Delta y = -y - w$ and $\Delta s = -s - b - B(y + \Delta y)$. That is, we must be able to solve equations involving the matrices C and $[BC^{-1}B^{\mathrm{T}} + D(s/u)]$. The Sherman-Morrison-Woodbury formula can be applied to invert $[BC^{-1}B^{\mathrm{T}} + D(s/u)]$ giving the alternative representation

$$\begin{aligned}&[BC^{-1}B^{\mathrm{T}} + D(s/u)]^{-1} = D(s/u)^{-1} \\ &-D(s/u)^{-1}B[C + B^{\mathrm{T}}D(s/u)^{-1}B]^{-1}B^{\mathrm{T}}D(s/u)^{-1}\end{aligned} \tag{22}$$

Hence, the problem reduces to being able to solve equations involving matrices of the form $C + B^{\mathrm{T}}DB$, where D is either zero or positive definite and diagonal. This can be accomplished using the following theorem.

Theorem 3 *Suppose that the matrix C is given by equation (12). It follows that the matrix $C + B^{\mathrm{T}}D(s/u)^{-1}B$ is symmetric positive definite, and has the same tridiagonal block structure as the matrix C. Further suppose that the sequence of derivatives $g_k^{(1)}(x_{k-1})$ for $k = 2, \ldots, N$ are invertible matrices. It follows that, C is positive definite and that, Δu in equation (21) and Δy in equation (20), can be calculated in a numerically stable fashion with $O(n^3 N)$ floating point operations using the block tridiagonal algorithm in [2].*

Proof: The matrix $B^{\mathrm{T}}D(s/u)^{-1}B$ is symmetric block diagonal with blocks of size $n \times n$. Hence $C + B^{\mathrm{T}}D(s/u)^{-1}B$ has the same tridiagonal block structure as the matrix C. The matrix C is the Hessian, with respect to $\{y_k\}$, of

$$\tilde{S}(\{x_k\}; \{y_k\}) = \sum_{k=1}^{N} \tilde{S}_k(x_k, x_{k-1}; y_k, y_{k-1})$$

and the Hessian, with respect to $\{y_k\}$, of each $\tilde{S}_k(x_k, x_{k-1}; y_k, y_{k-1})$ is positive semi-definite since it is a positive semi-definite quadratic form (see equation (7) where both $\tilde{g}_k(x_k; y_k)$ and $\tilde{h}_k(x_k; y_k)$ are affine with respect to y_k). Hence C is symmetric and positive semi-definite. Given the structure of the blocks in C, displayed in equation (12), C satisfies the conditions in [2, Lemma 6] and we can compute Δy in equation (20). Hence C is nonsingular and so positive definite. Since each of the $n \times n$ diagonal blocks in $B^{\mathrm{T}}D(s/u)^{-1}B$ is positive semi-definite, $C + B^{\mathrm{T}}D(s/u)^{-1}B$ must also be positive definite. One can now apply equation (22) and [2, Lemma 6] to solve for Δu in equation (21). □

5 The Constrained QP Algorithm

In the case where the functions $\{f_k\}$, $\{g_k\}$, and $\{h_k\}$ are all affine (their derivatives are constant), the original problem (5) is equivalent to the QP (14). The algorithm below terminates at an approximate solution to QP (14) consisting of a primal vector $y^p \in \mathbf{R}^{nN}$ and Lagrange multiplier vector $u^p \in \mathbf{R}_+^{\ell N}$ satisfying

$$\begin{aligned}&b + By^p \le \delta\ ,\ \|B^{\mathrm{T}}u^p + Cy^p + d\|_\infty \le \delta\ ,\ \text{and} \\ &\|u^p \cdot (b + By^p)\|_\infty \le \delta\end{aligned} \tag{23}$$

where p is the iteration index and for $v \in \mathbf{R}^q$ and $w \in \mathbf{R}^q$, $v \le \delta$ means $v_i \le \delta$ for $i = 1, \ldots, q$, $v \cdot w \in \mathbf{R}^q$ is defined by $(v \cdot w)_i = v_i w_i$, and $\|v\|_\infty$ is maximum of $|v_i|$ for $i = 1, \ldots, q$. The convergence criteria (23) correspond to a δ relaxed version of the KKT conditions for QP (14). These KKT conditions are given by equation (16) with $\mu = 0$. To be more specific, the first term in criteria (23) corresponds to a δ relaxation of the feasibility condition $b + By \le 0$, the second term corresponds to a δ relaxation of the Lagrangian stationarity condition $\nabla_y L_\mu(y, s, u) = 0$, and the third term correspond to a δ relaxation of the complementarity condition $u \cdot (b + By) = 0$.

The algorithm below solves the QP (14). It is a naive implementation of the interior point method. In the algorithm, scalar and function iterates are subscripted by the iteration counter p while vector iterates are superscripted by p. Given a vector w, $\max(w)$ denotes the maximum component of w, and $\|w\|_2$ denotes the square root of the sum of squares of the components of w.

The following is a list of the inputs to the algorithm below: the convergence tolerance $\delta > 0$, the quadratic factor in the objective $C \in \mathbf{R}^{nN \times nN}$, the linear factor in the objective $d \in \mathbf{R}^{nN}$, the constant term in the inequality constraint $b \in \mathbf{R}^{\ell N}$, and the linear factor in the inequality constraint $B \in \mathbf{R}^{\ell N \times nN}$.

Algorithm 4 *Inequality Constrained Affine Smoother*

(1) Initialization: Let $y^0 \in \mathbf{R}^{nN}$ be the solution to the unconstrained problem; i.e., $y^0 = -C^{-1}d$. If $\max(b + By^0) \le \delta$, set $u^0 = 0$ and return (y^0, u^0) (the convergence criteria (23) are satisfied by this pair). Otherwise set $p = 0$, $\mu_0 = \max(b + By^0)$, and $s^0 = u^0 = \sqrt{\mu_0}e$.

(2) Solve the following equation for Δs^p, Δy^p, Δu^p

$$\begin{aligned}0 \;=\; &F_{\mu(p)}(s^p, y^p, u^p) \\ &+F^{(1)}_{\mu(p)}(s^p, y^p, u^p)(\Delta s^p\ ,\ \Delta y^p\ ,\ \Delta u^p)^{\mathrm{T}}\end{aligned}$$

(3) Compute the largest step size β_p less than or equal to 1 such that $s^p + \beta_p \Delta s^p \ge s^p/10$ and $u^p + \beta_p \Delta u^p \ge u^p/10$. This is calculated as follows

$$\begin{aligned}\alpha_p =\ &\max\left(\max(-\Delta s^p/s^p), \max(-\Delta u^p/u^p)\right) \\ \beta_p = &\begin{cases} 1 & \text{if } \alpha_p \le 0.9 \\ 0.9/\alpha_p & \text{otherwise} \end{cases}\end{aligned}$$

(4) Compute the line search parameter λ_p, new iterates s^{p+1}, y^{p+1}, u^{p+1}, and penalty parameter μ_{p+1}

$$G_p(\lambda) = \|F_{\mu(p)}(s^p + \lambda\Delta s^p\ ,\ y^p + \lambda\Delta y^p\ ,\ u^p + \lambda\Delta u^p)\|_2^2$$
$$\lambda_p = \max\{\ \beta_p 2^{-q} \mid q \in \mathbf{Z}_+ \text{ and } G_p(\beta_p 2^{-q}) \le (1 - \beta_p 2^{-q}/10) G_p(0)\ \}$$

where $\mathbf{Z}_+$ is the nonnegative integers. Set $s^{p+1} = s^p + \lambda_p \Delta s^p$, $y^{p+1} = y^p + \lambda_p \Delta y^p$, and $u^{p+1} = u^p + \lambda_p \Delta u^p$. If $\mathrm{mod}(p, 3) = 1$ then $\mu_{p+1} = \mu_p$ else $\mu_{p+1} = \mu_p/10$.

(5) Set $p = p + 1$. If the convergence criteria (23) are satisfied, return y^p, u^p; otherwise go to step 2.

6 The Unconstrained Nonlinear Algorithm

As an introduction to Algorithm 6 below, we first present an unconstrained version. The unconstrained version of problem (5) is

$$\text{minimize } S(\{x_k\}) \text{ w.r.t. } \{x_k\}$$

We solve this problem by iteratively solving different versions of an approximating QP. This is similar to the Gauss-Newton method in [1], except that a line search is included to ensure global convergence; see Theorem 9. The unconstrained QP that approximates the problem above near $\{x_k\}$ is minimize $\tilde{S}(\{x_k\}; \{y_k\})$ with respect to $\{y_k\}$ which is equivalent to

$$\text{minimize } \tfrac{1}{2} y^{\mathrm{T}} C y + d^{\mathrm{T}} y \text{ w.r.t. } y \in \mathbf{R}^{nN} \tag{24}$$

where C and d are the same as in problem (14); i.e., a is defined by equation (13), C is defined by equation (12), $x \in \mathbf{R}^{nN}$ is the column vector representing $\{x_k\}$, and $d = a - Cx$. The solution of the QP (24) is $y = -C^{-1}d$. This can be computed in a numerically stable fashion, with $O(n^3 N)$ floating point operations, using the block tridiagonal algorithm in [2]. The unconstrained first order convergence criteria is

$$\|\partial_{x(k)} S(\{x_j^p\})\|_\infty \le \varepsilon \qquad (k = 1, \cdots, N) \tag{25}$$

The inputs to the unconstrained algorithm are as follows (values with subscript k are known for all k): the convergence tolerance ε, the dynamical model g_k in equation (1), the measurement model h_k in equation (1), the measurement values z_k in equation (1), the dynamics covariance Q_k in equation (2), the measurement covariance R_k in equation (2), and the starting state vector for the optimization process $x_k^0 \in \mathbf{R}^n$.

Algorithm 5 *Unconstrained Nonlinear Smoother*

(1) Initialization: Set the iteration counter $p = 0$.
(2) Affine approximation: Substitute $\{x_k^p\}$ for $\{x_k\}$ in equations (12), (13) and let C^p and d^p be the corresponding values for C, d in QP (24).
(3) Let $y^p = -(C^p)^{-1} d^p$.
(4) If the convergence criteria (25) is satisfied, return x^p as the solution.
(5) Compute the line search step size λ_p as follows:

$$\eta_p = (a^p)^{\mathrm{T}}(y^p - x^p)\ ,\ H_p(\lambda) = S[x^p + \lambda(y^p - x^p)]$$
$$\lambda_p = \max\{\ 2^{-q} \mid q \in \mathbf{Z}_+ \text{ and } H_p(2^{-q}) - H_p(0) \le 2^{-q}\eta_p/10\ \}$$

(6) Set $x^{p+1} = x^p + \lambda_p(y^p - x^p)$, then set $p = p+1$ and go to step 2.

7 The Constrained Nonlinear Algorithm

Given a termination tolerance ε, the termination criteria for the general nonlinear problem are based on the KKT conditions for problem (5). Specifically, the algorithm terminates at a primal vector $x^p \in \mathbf{R}^{nN}$ and Lagrange multiplier vector $u^p \in \mathbf{R}_+^{\ell N}$ such that for $k = 1, \ldots, N$

$$\begin{aligned} &f_k(x_k^p) \le \varepsilon\ ,\ \|u_k^p \cdot f_k(x_k^p)\|_\infty \le \varepsilon \text{ and} \\ &\|(u_k^p)^{\mathrm{T}} f_k^{(1)}(x_k^p) + \partial_{x(k)} S(\{x_j^p\})\|_\infty \le \varepsilon \end{aligned} \tag{26}$$

where the sequence $\{x_k^p\}$ with each $x_k^p \in \mathbf{R}^n$ corresponds to the vector $x^p \in \mathbf{R}^{nN}$ and $\{u_k^p\}$ with each $u_k^p \in \mathbf{R}^\ell$ corresponds to the vector $u^p \in \mathbf{R}^{\ell N}$.

Given a vector $w \in \mathbf{R}^q$, we use $\max(0, w) \in \mathbf{R}^q$ to denote the vector with i-th component equal to $\max(0, w_i)$. Given a $x \in \mathbf{R}^{nN}$, the $\boldsymbol{\ell}_1$ distance from the constraint function values $\{f_k(x_k)\}$ to the constraint set is

$$\phi(x) = \sum_{k=1}^{N} \sum_{i=1}^{\ell} \max[f_k(x_k)_i, 0]$$

and its approximation is given by

$$\tilde{\phi}(x; y) = \sum_{k=1}^{N} \sum_{i=1}^{\ell} \max[\tilde{f}_k(x_k; y_k)_i, 0]$$

The algorithm below solves the nonlinear problem (5). The inputs to this algorithm are the same as for unconstrained algorithm in Section (6) with the addition of the constraint functions $f_k : \mathbf{R}^n \to \mathbf{R}^\ell$ for $k = 1, \ldots, N$.

Algorithm 6 *Inequality Constrained Nonlinear Smoother*

(1) Initialization: Set the iteration counter $p = 0$ and the initial penalty parameter $\alpha_0 = 0$.

(2) Affine approximation: Substitute $\{x_k^p\}$ for $\{x_k\}$ in equations (12), (13) and let a^p, b^p, B^p, C^p, and d^p be the corresponding values for a, b, B, C, and d in QP (14).
(3) Solve this QP using Algorithm 4 with inputs $\delta = \varepsilon \times 10^{-2}$, $C = C^p$, $d = d^p$, $b = b^p$, and $B = B^p$. Let y^p and u^p be the resulting solution.
(4) If the convergence criteria (26) are satisfied, return x^p, u^p as the solution.
(5) If $\alpha_p > 0$, set $\hat{\alpha}_p = \alpha_p$; otherwise, $\hat{\alpha}_p = \|u^p\|_\infty$. Define the value

$$\zeta_p = (y^p - x^p)^{\mathrm{T}} C^p (y^p - x^p) + (a^p)^{\mathrm{T}} (y^p - x^p)$$

If $\zeta_p \leq \hat{\alpha}_p \phi(x^p)$, set $\alpha_{p+1} = \hat{\alpha}_p$; otherwise, $\alpha_{p+1} = \max[\zeta_p / \phi(x^p)\ ,\ 2\hat{\alpha}_p]$.
(6) Compute the line search step size λ_p as follows:

$$\begin{aligned}
\eta_p &= (a^p)^{\mathrm{T}}(y^p - x^p) + \alpha_{p+1}[\tilde{\phi}(x^p; y^p) - \phi(x^p)] \\
H_p(\lambda) &= S[x^p + \lambda(y^p - x^p)] \\
&\quad + \alpha_{p+1}\phi[x^p + \lambda(y^p - x^p)] \\
\lambda_p &= \max\{\ 2^{-q} \mid q \in \mathbf{Z}_+ \text{ and} \\
&\quad H_p(2^{-q}) - H_p(0) \leq 2^{-q}\eta_p/10\ \}
\end{aligned}$$

(7) Set $x^{p+1} = x^p + \lambda_p(y^p - x^p)$, then set $p = p+1$ and go to step 2.

Remark 7 *This algorithm can be generalized using the model algorithm in [5]. One natural extension considers the case where the subproblems do not have feasible solutions. In addition, the convergence theory in [5] can be extended to the case where the subproblems are not solved exactly; i.e., $\delta > 0$; see [6, Section 7, Remark 6].*

Lemma 8 *Suppose $\varepsilon = 0$, all the quadratic subproblems in step 2 have feasible solutions, and the corresponding sequence $\{y^p\}$ is bounded. Then Algorithm 6 is a special case of the model algorithm in [5, Section 4], the sequences $\{x^p\}$ and $\{C^p\}$ are bounded, and $\{C^p\}$ is contained in a compact set of real symmetric positive definite matrices. (See proof in Appendix.)*

The convergence result in Theorem 9 below requires the use of the constraint qualification given in [5, Definition 5.1]. This constraint qualification is a modest extension of the well-known Mangasarian-Fromowitz constraint qualification (MFCQ) [19, Definition 12.6]. The extension allows the application of the condition to infeasible as well as feasible points, and is equivalent to the MFCQ at feasible points. For this reason we refer to this extension as the Mangasarian-Fromowitz constraint qualification. However, we mention that there are many possible extensions of the Mangasarian-Fromowitz constraint qualification to non-feasible points. For example, generalizations that exploit the geometry induced by a specific choice of norm are given in [6, Definition 9.1].

MFCQ: Problem (5) can be written as

$$\text{minimize } S(x) \text{ w.r.t. } x \in \mathbf{R}^{nN} \text{ subject to } f(x) \leq 0$$

A point $\hat{x} \in \mathbf{R}^{nN}$ satisfies the Mangasarian-Fromowitz Constraint Qualification for this problem if there is a direction $\hat{w} \in \mathbf{R}^{nN}$ such that if $i \in \{1, \ldots, \ell N\}$ and $f_i(\hat{x}) \geq 0$ then $f_i^{(1)}(\hat{x})\hat{w} < 0$.

A point $\hat{x}$ is a cluster point of the sequence $\{x^p\}$ if for every $\varepsilon > 0$ there are infinitely many indices p such that $|x^p - \hat{x}| < \varepsilon$.

Theorem 9 *Suppose $\varepsilon = 0$, all the quadratic subproblems in step 2 have feasible solutions, the corresponding sequence $\{y^p\}$ is bounded, and every cluster point of $\{x^p\}$ satisfies the MFCQ. Then the sequence $\{x^p\}$ is bounded and each of its cluster points is a KKT point for problem (5), i.e. satisfies convergence criteria (26) for some vector of Lagrange multipliers. (See proof in Appendix.)*

8 Affine Example

We consider the problem of estimating an unknown function from a finite set of noisy measurements of the function's value. For the purpose of simulating the measurements $\{z_k\}$, the function's derivative $X_1(t)$ and its value $X_2(t)$ are given by

$$X(t) = [\ -\cos(t)\ ,\ \ -\sin(t)\ ]^{\mathrm{T}}$$

Our model for this unknown function is given by the stochastic differential equation (see [20] or [10])

$$\mathbf{d}X(t) = FX(t)\ \mathbf{d}t + G\ \mathbf{d}B(t) \tag{27}$$

where $B(t)$ is Brownian motion and

$$F = \begin{pmatrix} 0 & 0 \\ 1 & 0 \end{pmatrix} \quad , \quad G = \begin{pmatrix} 1 \\ 0 \end{pmatrix}$$

Thus the $X_1(t)$ is equal to $B(t)$ plus a constant of integration and $X_2(t)$ is just the integral of $X_1(t)$. This model is often used to describe signals that are known to be smooth (but nothing more). In particular, it forms the basis for a Bayesian interpretation of cubic smoothing splines. The problem of reconstructing $X_2(t)$ from a finite set of noisy direct measurements, subject to linear inequality constraints, is considered in [29, Section 9.4]. The interior point Kalman smoother presented above solves this problem efficiently with the number of operations scaling linearly with the number of observation time points.

For $s \leq t$, let $P(t|s)$ denote the covariance of $X(t)$ given the value of $X(s)$. In general, when F, G are constant

matrices and $B(t)$ is Brownian motion with identity matrix covariance per unit time, the covariance $P(t|s)$ satisfies a differential Lyapunov equation (see [14, page 133 equation 4.138]). To be more specific, the covariance of the state at time t, conditioned on state value at time s, and prior to knowing the system output measurements, satisfies the differential Lyapunov equation

$$P(s|s) = 0 \text{ and } \partial_t P(t|s) = FP(t|s) + P(t|s)F^{\mathrm{T}} + GG^{\mathrm{T}}$$

For our particular choice of F and B,

$$\begin{aligned}
\partial_t P(t|s) &= \begin{pmatrix} 0 & 0 \\ 1 & 0 \end{pmatrix} P(t|s) + P(t|s) \begin{pmatrix} 0 & 1 \\ 0 & 0 \end{pmatrix} + \begin{pmatrix} 1 & 0 \\ 0 & 0 \end{pmatrix} \\
&= \begin{pmatrix} 1 & P_{1,1}(t|s) \\ P_{1,1}(t|s) & P_{1,2}(t|s) + P_{2,1}(t|s) \end{pmatrix} \\
P(t|s) &= \begin{pmatrix} (t-s) & (t-s)^2/2 \\ (t-s)^2/2 & (t-s)^3/3 \end{pmatrix}
\end{aligned}$$

In this example, Δt is constant between time points. Thus, the dynamical model for $k > 1$ is represented by $g_k : \mathbf{R}^2 \to \mathbf{R}^2$ and $Q_k \in \mathbf{R}^{2\times 2}$ where

$$\begin{aligned}
g_k(x_{k-1}) &= (x_{1,k-1}\ ,\ x_{2,k-1} + x_{1,k-1}\Delta t)^{\mathrm{T}} \\
Q_k &= \begin{pmatrix} \Delta t & \Delta t^2/2 \\ \Delta t^2/2 & \Delta t^3/3 \end{pmatrix}
\end{aligned}$$

The initial state estimate is given by $g_1(x_0) = X(t_1)$ and $Q_1 = 100 I_2$ where I_2 is the two by two identity matrix. The covariance of the initial state estimate is large so that it has no noticeable affect on the resulting fit. The measurement variance σ^2 is also constant. Direct measurements of $X_2(t)$ are represented by $h_k : \mathbf{R}^2 \to \mathbf{R}^1$ and $R_k \in \mathbf{R}^{1\times 1}$ where

$$h_k(x_k) = x_{2,k} \quad , \quad R_k = \sigma^2$$

The function $X(t)$ satisfies $-1 \le X_1(t) \le 1$ and $-1 \le X_2(t) \le 1$. We require that the estimate of $X(t)$ satisfy the same constraints which we represent by $f_k(x_k) \le 0$ where $f_k : \mathbf{R}^2 \to \mathbf{R}^4$ is defined by

$$f_k(x_k) = (-x_{1,k} - 1,\ x_{1,k} - 1,\ -x_{2,k} - 1,\ x_{2,k} - 1)^{\mathrm{T}}$$

The specifications for the example are completed by : the number of measurement times $N = 50$, the spacing between time points $\Delta t = 2\pi/N$, the time corresponding to k-th measurement $t_k = k\Delta t$, the standard deviation of the measurement noise $\sigma = .5$, the simulated measurement noise $v_k \sim \mathbf{N}(0, R_k)$, and the simulated measurement values $z_k = h_k[X(t_k)] + v_k$.

The results of fitting a typical realization are plotted in Figure 1 where circles denote the noisy measurements

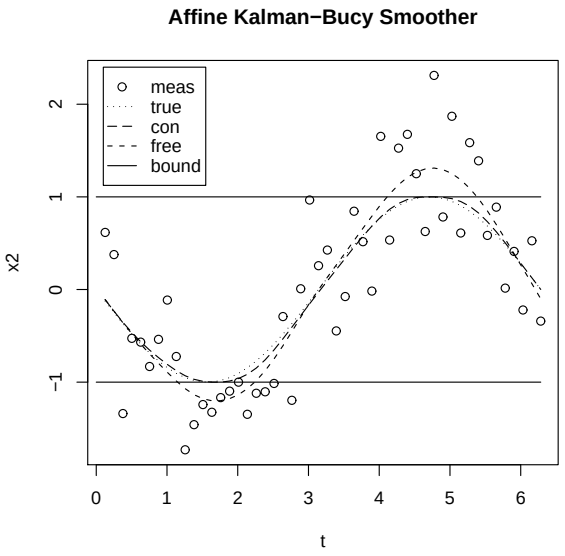

Fig. 1. Measurement values (meas), simulated function (true), constrained estimate (con), unconstrained estimate (free), and estimate bounds (bound)

and the solid lines denote the constraints. We can see the beneficial effect of the constraints: notice that the constrained estimate (long dashed-line) is much closer to the true function (dotted line) when compared with the unconstrained estimate (short dashed-line). The unconstrained estimate is the initial state sequence used during the optimization process to determine the constrained estimate; i.e., y^0 in step 1 of Algorithm 4. We also note that this initial state sequence is not feasible.

9 Nonlinear Example

Consider the problem of tracking a ship traveling close to shore where we are given distance measurements from two fixed stations to the ship as well as the location of the shoreline. The corresponding measurement functions $\{h_k\}$ are not affine because the distance is not affine with respect to the position of the ship. In addition, the corresponding constraint functions $\{f_k\}$ are not affine because the shoreline is not a straight line. For the purpose of simulating the measurements $\{z_k\}$, the ship velocity $[X_1(t), X_3(t)]$ and the ship position $[X_2(t), X_4(t)]$ are given by

$$X(t) = [\ 1\ ,\ t\ ,\ -\cos(t)\ ,\ 1.3 - \sin(t)\ ]^{\mathrm{T}}$$

We model the ship velocity components as independent Brownian motions plus an initial velocity, and the position of the ship as the integral of the velocity plus an initial location. This example also uses a constant spacing Δt between time points. The dynamical model for

$k > 1$ is given by $g_k : \mathbf{R}^4 \to \mathbf{R}^4$, $Q_k \in \mathbf{R}^{4\times 4}$ where

$$g_k(x_{k-1}) = (x_{1,k-1} \,,\; x_{2,k-1} + x_{1,k-1}\Delta t \,,\; x_{3,k-1} \,,\; x_{4,k-1} + x_{4,k-1}\Delta t)^{\mathrm{T}}$$

$$Q_k = \begin{pmatrix} \Delta t & \Delta t^2/2 & 0 & 0 \\ \Delta t^2/2 & \Delta t^3/3 & 0 & 0 \\ 0 & 0 & \Delta t & \Delta t^2/2 \\ 0 & 0 & \Delta t^2/2 & \Delta t^3/3 \end{pmatrix}$$

The initial state estimate is given by $g_1(x_0) = X(t_1)$ and $Q_1 = 100 I_4$ where I_4 is the four by four identity matrix. The measurement variance is constant for this example and is denoted by σ^2. The distance measurements are made from two stationary locations on shore. One is located at $(0, 0)^{\mathrm{T}}$ and the other is located at $(2\pi, 0)^{\mathrm{T}}$. These distance measurements are represented by $h_k : \mathbf{R}^4 \to \mathbf{R}^2$ and $R_k \in \mathbf{R}^{2\times 2}$ where

$$h_k(x_k) = \begin{pmatrix} \sqrt{x_{2,k}^2 + x_{4,k}^2} \\ \sqrt{(x_{2,k} - 2\pi)^2 + x_{4,k}^2} \end{pmatrix} \;,\; R_k = \begin{pmatrix} \sigma^2 & 0 \\ 0 & \sigma^2 \end{pmatrix}$$

We know that the ship does not cross land and so $X_4(t) \geq 1.25 - \sin[X_2(t)]$. This information is represented by $f_k(x_k) \leq 0$ where $f_k : \mathbf{R}^4 \to \mathbf{R}^1$ is defined by

$$f_k(x_k) = 1.25 - \sin(x_{2,k}) - x_{4,k}$$

The specifications for this example are completed by : the number of measurement times $N = 50$, the spacing between time points $\Delta t = 2\pi/N$, the time corresponding to k-th measurement $t_k = k\Delta t$, the standard deviation of the measurement noise $\sigma = .25$, the simulated measurement noise $v_k \sim \mathbf{N}(0, R_k)$, and the simulated measurement values $z_k = h_k[X(t_k)] + v_k$.

Our choice for the initial state sequence for the optimization process is given by $x_{1,k}^0 = 0$, $x_{2,k}^0 = 0$, $x_{3,k}^0 = 0$, and $x_{4,k}^0 = 1$. It follows that, for $k = 1, \ldots, N$, the corresponding value for the constraint function is

$$f_k(x_k^0) = 1.25 - \sin(x_{2,k}) - x_{4,k} = .25 > 0$$

Thus, the initial state vector for this example problem is not feasible. The results of fitting a typical realization are now plotted in Figure 2 with the same conventions as adopted in Figure 1 (except that measurements are not reported in Figure 2). The beneficial effect of including the constraints similar to those in Figure 1.

10 Conclusions

In many situations, certain information about a dynamical system is most naturally represented using

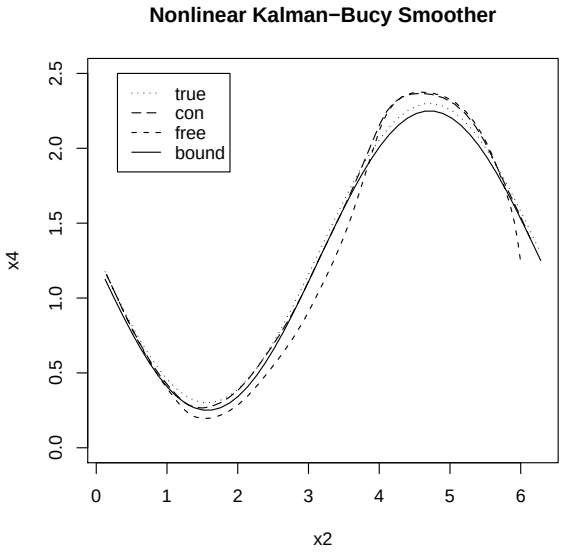

Fig. 2. Simulated function (true), constrained estimate (con), and unconstrained estimate (free), estimate bound (bound)

constraints. Including such information can compensate for model approximation errors and greatly improve the state estimates. Strategies for including equality constraints can be found in the literature. However, handling inequality constraints is a much harder problem due to the complexity of identifying those constraints that are active at the solution (e.g. see [9, equation (25)]). Nonetheless, estimates that respect inequality constraints may be crucial; e.g., [3,9,11,18,21,24,27,28,34]. In this paper we have shown how interior point methods can be applied to maximize the Kalman-Bucy smoother likelihood subject to nonlinear inequality constraints. A key contribution of this approach is that it exploits the same decomposition that is used for unconstrained Kalman-Bucy smoothers and so the required operations scale linearly with the number of measurements. Interior point methods allow for efficient and highly accurate solution of the QP subproblem (14). Possible future enhancements may be obtained by applying more sophisticated techniques from the interior point literature and by improving robustness as suggested in Remark 7.

Matlab code and documentation for the algorithms can be found at the address below. If your browser does not support MathML, use `ckbs.xml` instead of `ckbs.xml`.
`http://www.seanet.com/~bradbell/ckbs/ckbs.xml`

Appendix

Proof of Lemma 8: Let θ be a bound for x^0 and the sequence $\{y^p\}$; i.e., $\|x^0\|_2 \leq \theta$ and $\|y^p\|_2 \leq \theta$. We show by induction that $\|x^p\| \leq \theta$ for all p. It is true for $p = 0$ so assume it is true for some index p. It follows from step 7 that x^{p+1} is a convex combination of x^p and y^p. It now

follows that $\|x^{p+1}\|_2 \leq \theta$. This completes the induction and shows that the sequence $\{x^p\}$ is bounded.

In what follows, we use Model algorithm to refer to the model algorithm in [5, Section 4]. We use $\|w\|_1$ to denote the $\boldsymbol{\ell}_1$ norm of w; i.e., $\|w\|_1 = \sum_{i=1}^{q} |w_i|$ The table below lists the notational correspondence between the Model algorithm in [5] and Algorithm 6. In the table, P_{α_i} is an abbreviation for $P_{\alpha_i}(x_i + \lambda d_i)$ and $f(x)_+$ is an abbreviation for $\max[f(x), 0]$.

Model	**Algo 6**	**Description**
x_i	x^p	state sequence for this iterate
α_i	α_p	penalty parameter
$f(x_i)$	$S(x^p)$	objective function
$g(x_i)$	$f(x^p)$	inequality constraint function
$h(x_i)$	none	equality constraint function
$\nabla f(x_i)$	a^p	gradient of objective
H_i	C^p	subproblem quadratic factor
$r_1(x_i)$	0	constraint feasibility factor
$\|w\|$	$\|w\|_1$	the $\boldsymbol{\ell}_1$ norm of w
$\|w\|_0$	$\|w\|_\infty$	the $\boldsymbol{\ell}_\infty$ norm of w
$R_0(x)$	$f(x)_+$	constraint violation
β_i	β	subproblem trust region radius
$-\Delta(x_i, \sigma_i)$	$\phi(x^p)$	$\boldsymbol{\ell}_1$ norm of constraint violation
d_i	$y^p - x^p$	line search direction
P_{α_i}	$H_p(\lambda)$	penalty function
λ_i	λ_p	line search parameter

By equation (10), and the definition of y^p in step 3 of Algorithm 6, $w^p = y^p - x^p$ solves the QP

$$\begin{array}{ll} \text{minimize} & \frac{1}{2} w^{\mathrm{T}} C w + S^{(1)}(x^p) w \quad \text{w.r.t. } w \in \mathbf{R}^{nN} \\ \text{subject to} & f(x^p) + f^{(1)}(x^p) w \leq 0 \end{array}$$

The sequence $\{w^p\}$ is bounded, because both $\{y^p\}$ and $\{x^p\}$ are bounded. Thus w^p also solves the QP above with the additional constraint $\beta \geq \|w\|_1$ for some sufficiently large and fixed value of β. Therefore the problems in step 3 are instances of the QP subproblems referred to as $Q(x_i, H_i, \sigma_i, \beta_i)$ in the Model algorithm.

The boundedness of $\{x^p\}$, implies that there is a compact set $\Omega \subset \mathbf{R}^{nN}$ such that $x^p \in \Omega$ for all p. We define the mapping $\omega : \Omega \to \mathbf{R}^{nN \times nN}$ by $\omega(x)$ is the matrix C that corresponds to x in equation (12). It follows from Theorem 3 that each $C \in \omega(\Omega)$ is positive definite. It follows from the continuity of $\omega(x)$ that $\omega(\Omega)$ is compact. Thus, the sequence $\{C^p\}$ is contained in the compact set of real symmetric positive definite matrices $\omega(\Omega)$.

In step 5 the value $\hat{\alpha}_p$ is used to initialize the first non-zero value of α_p. To be specific, the first non-zero multiplier vector u^p is used to set the first non-zero value of α_p. It follows from $\varepsilon = 0$, that $\delta = 0$ which implies that the subproblem is solved exactly; hence $\tilde{\phi}(x^p, y^p) = 0$. It now follows that the formulas for α_{p+1} are the same as in step (3) of the Model algorithm.

Using the fact that $\delta = 0$ and y^p solves the QP in step 3, we conclude that $\tilde{\phi}(x^p; y^p) = 0$ and in step 6 we have

$$\eta_p = a^p(y^p - x^p) - \alpha_{p+1}\phi(x^p)$$

As a special case of the Model algorithm, we can choose its values $\gamma_1 = 1$, $\gamma_2 = 1/2$, and $\mu_1 = \mu_2 = 1/10$. Note that if $\lambda_p < \gamma_1$,

$$H_p(1) - H_p(0) > \mu_2 \eta_p$$

It follows that the procedure for choosing λ_p in step 6 is an instance of step (4) of the Model algorithm (where $\bar{\lambda}_i = \lambda_i/\gamma_2$ in step (4) of the Model algorithm). Therefore, Algorithm 6 is a special case of the Model algorithm. □

Proof of Theorem 9: The hypotheses of Lemma 8 are satisfied, hence the sequence $\{x^p\}$ is bounded. Let $\hat{x}$ be a cluster point for $\{x^p\}$ and let $J \subset \mathbf{Z}_+$ be a subsequence for which $x^p \overset{J}{\to} \hat{x}$. The point $\hat{x}$ can not be a Fritz John point (in the sense of [5]) because the MFCQ is satisfied at $\hat{x}$. We note that for all x, $S(x) \geq 0$, $\phi(x) \geq 0$. It follows that for all p and λ, $H_p(\lambda) \geq 0$. Hence, [5, Theorem 6.1, Case (2)-(a)] is not possible.

Case 1: Suppose that $\hat{x}$ is not feasible. It follows that it cannot be a stationary point for the penalty function $\phi(x)$ [5, Theorem 5.1, Case (2)]. It now follows from [5, Corollary 6.1] that the sequence of penalty parameters $\{\alpha_p\}$ is bounded. It follows that [5, Theorem 6.1, Case (1)] and [5, Theorem 6.1, Case (2)-(b)] are not possible. Thus, by [5, Theorem 6.1, Case (2)-(c)] we conclude that $\hat{x}$ must be feasible.

Case 2: Suppose the sequence $\{\alpha_p\}$ is not bounded. This sequence is monotone increasing and hence there must be a cluster point $\hat{x}$ for the set $\{x^p : \alpha_p < \alpha_{p+1}\}$. It follows from [5, Corollary 6.1] that $\hat{x}$ is a not feasible, but this contradicts the conclusion for Case 1 above.

Case 3: Since neither Case 1 or Case 2 can occur, we conclude that $\hat{x}$ is feasible and the sequence $\{\alpha_p\}$ is bounded. It follows that [5, Theorem 6.1, Case (1)] and [5, Theorem 6.1, Case (2)-(b)] are not possible. Thus, by [5, The-

orem 6.1, Case (2)-(c)] we conclude that $(y^p - x^p) \to 0$, $\phi(x^p) \to 0$, and

$$S^{(1)}(x^p) + (w^p)^{\mathrm{T}} f^{(1)}(x^p) \to 0$$

where w^p is any Lagrange multiplier for the corresponding QP in step 3. Any multiplier for the subproblem must satisfy the corresponding complementarity condition

$$\|w^p \cdot \tilde{f}(x^p; y^p)\|_\infty = 0$$

It follows from [5, Theorem 5.1, Item (4)] that we can choose the multipliers w^p to be bounded in a neighborhood of $\hat{x}$. There is a further subsequence of indices $K \subset J$, and a multiplier value $\hat{w}$, such that $x^p \overset{K}{\to} \hat{x}$ and $w^p \overset{K}{\to} \hat{w}$. It follows that $\tilde{f}(x^p; y^p) \overset{K}{\to} f(\hat{x})$, $f(\hat{x}) \leq 0$, $\|\hat{w}^{\mathrm{T}} f^{(1)}(\hat{x}) + S^{(1)}(\hat{x})\|_\infty = 0$, and $\|\hat{w} \cdot f(\hat{x})\|_\infty = 0$; i.e., $\hat{x}$ is a KKT point for problem (5). □

References

[1] Bell, B.M., The iterated Kalman smoother as a Gauss-Newton method *SIAM Journal on Optimization*, 4(3): 626-636, 1994.

[2] Bell, B.M., The marginal likelihood for parameters in a discrete Gauss-Markov process *IEEE Transactions on Signal Processing*, 48: 870-873, 2000.

[3] Boers, Y., H. Driessen, Particle filter track-before-detect application using inequality constraints, *IEEE Transactions on Aerospace and Electronic Systems*, 41(4): 1481-1487, 2005.

[4] Burke, J.V., J. J. Moré, On the identification of active constraints, *SIAM J. Numer. Anal.*, 25(5): 1197-1211, 1988.

[5] Burke, J.V., S-P. Han, A robust sequential quadratic programming method, *Mathematical Programming*, 43(1-3): 277-303, 1989.

[6] Burke, J.V., A sequential quadratic programming method for potentially infeasible mathematical programs, *J. Math. Anal, & Appl.*, 139: 319-351, 1989.

[7] Burke, J.V., J. J. Moré, Exposing constraints, *SIAM J. Control and Optimization*, 4(3): 573-595, 1994.

[8] Chung, S., C. Kuo, P. Shih, Rate-constrained motion estimation using Kalman filter, *Journal of Visual Communication and Image Representation*, 17(4): 929-946, 2006.

[9] Darby, M.L., M. Nikolaou, A parametric programming approach to moving-horizon state estimation, *Automatica*, 43(5): 885-891, 2007.

[10] De Nicolao, G., G. Ferrari Trecate, Regularization networks for inverse problems: a state space approach, *Automatica*, 39: 669-676, 2003.

[11] El-Keyi, A., T. Kirubarajan, A.B. Gershman, Robust adaptive beamforming based on the Kalman filter, *IEEE Transactions on Signal Processing*, 53(8): 3032-3041, 2005.

[12] Facchinei, F., A. Fischer, C. Kanzow, On the accurate identification of active constraints, *SIAM Journal on Optimization*, 9(1): 14-32, 1998.

[13] A.V. Fiacco, G.P. McCormack, *Nonlinear Programming: Sequential Unconstrained Minimization Techniques*, *SIAM series Classics in Applied Mathematics*, Philadelphia, 1990.

[14] Jazwinski, A.H., *Stochastic Processes and Filtering Theory*, Academic Press, New York, 1970.

[15] Ko, S., R. Bitmead, State estimation for linear systems with state equality constraints, *Automatica*, 43(8): 1363-1368, 2007.

[16] Kojima, M., N. Megiddo, T. Noma, and A. Yoshise, *A unified approach to interior point algorithms for linear complementarity problems*, Lecture Notes in Computer Science 538, Springer-Verlag, Berlin, 1991.

[17] Lewis, A.S., Active sets, nonsmoothness and sensitivity, *SIAM Journal on Optimization*, 13(3): 702-725, 2002.

[18] Ma, N. M. Bouchard, R.A. Goubran, Speech enhancement using a masking threshold constrained Kalman filter and its heuristic implementations, *IEEE Transactions on Audio, Speech, and Language Processing*, 14(1): 19-32, 2006.

[19] Nocedal, J., S. J. Wright, *Numerical Optimization: 2nd Edition*, Springer-Verlag, New York, 2006.

[20] Oksendal, B., *Stochastic differential equations*, Springer, 1998.

[21] Rao, C.V., J.B. Rawlings, J.H, Lee, Constrained linear state estimation - A moving horizon approach, *Automatica*, 37(10): 1619-1628, 2001.

[22] Rao, C.V., J.B. Rawlings, D.Q. Mayne, Constrained state estimation for nonlinear discrete-time systems: stability and moving horizon approximations, *IEEE Transactions on Automatic Control*, 48(2): 246-258, 2003.

[23] Rauch, H.E., F. Tung, C. T. Striebel, Maximum likelihood estimates of linear dynamic systems, *J. Amer. Inst. Aeronautics and Astronautics*, 3(8): 1445-1450, 1965.

[24] Rawlings, J.B., B.R. Bakshi, Particle filtering and moving horizon estimation, *Computers and Chemical Engineering*, 30: 1529-1541, 2006.

[25] Rockafellar, R. T., *Convex Analysis*, Princeton University Press, Princeton, 1970.

[26] Simon, D., T.L. Chia, Kalman filtering with state equality constraints, *IEE Transcations on Aerospace and Electronic Systems*, 38(1): 128-136, 2002.

[27] Simon, D., D.L. Simon, Kalman filtering with inequality constraints for turbofan engine health estimation, *IEEE Proc-Control Theory and Applications*, 153(3): 371-378, 2006.

[28] Vachhani, P. Rengaswamy, R. V. Gangwal, S. Narasimhan, Recursive estimation in constrained nonlinear dynamical systems, *AIChE Journal (American Institute of Chemical Engineers)*, 51(3): 946-959, 2005.

[29] Wahba, G., *Spline Models for Observational Data*, SIAM, Philadelphia, 1990.

[30] Wang, L.S., Y.T. Chiang, F.R. Chang, Filtering method for nonlinear systems with constraints, *IEE Proceedings: Control Theory and Applications*, 149(6): 525-531, 2002.

[31] Woodbury, M. Inverting modified matrices, Technical Report Memorandum Report 42, Statistical Research Group, Princeton University, Princeton, NJ, 1950.

[32] Wright,
S.J., Identifiable surfaces in constrained optimization, *SIAM J. Control Optim.*, 31(4): 1063-1079, 1993.

[33] Wright, S.J., Interior point methods for optimal control of discrete time systems, *Journal of Optimization Theory and Applications*, 77(1): 161-187, 1993.

[34] Zaho, W., Z. Liu, R. Pei, Estimation of vehicle speed and friction force using moving horizon strategy, *Proceedings of the fifth World Congress on Intelligent Control and Automation*: 1547-1550, 2004.