Quadratic Programming for Monotone Control Theoretic Splines

Masaaki Nagahara
Graduate School of Informatics
Kyoto University
Kyoto, Japan
Email: nagahara@ieee.org

Clyde F. Martin
Department of Mathematics & Statistics
Texas Tech University
Texas, USA
Email: clyde.f.martin@ttu.edu

Yutaka Yamamoto
Graduate School of Informatics
Kyoto University
Kyoto, Japan
Email: yy@i.kyoto-u.ac.jp

***Abstract*—In this article, we solve the problem of monotone control theoretic splines. Monotone control theoretic splines have been solved only when the target system is the second-order integrator $1/s^2$, but not for other cases. To solve this, we first formulate the problem as semi-infinite quadratic programming, and then we adopt discretization technique. The problem is reduced to a finite-dimensional quadratic programming, which can be easily solved by numerical computation. We also illustrate examples to show the effectiveness of our method.**

I. Introduction

Control theoretic splines are interpolating smoothing splines with a constraint written in linear differential equations [4], that is, the interpolating curve between data points is obtained by an output of a given linear system. They have been proved to be useful in trajectory planning for aircrafts and robots, and statistics to smooth noisy data [3].

On the other hand, *monotone* control theoretic splines are also important in deriving a model or an estimation of a parameter such as the growing rate of an individual [2]. This needs to add a non-negative derivative constraint. The difficulty of this problem is that this constraint is infinite dimensional. In [2], this has been solved when the target linear system is a second-order integrator $1/s^2$. However, for general cases, there has been no solution.

In this article, we propose a new method for monotone splines with general linear systems. To solve this, we first formulate the problem as a semi-infinite quadratic programming. Then we adopt discretization technique [5] to reduce the infinite-dimensional constraints to a finite dimensional ones. This discretization technique is also know as fast discretization in sampled-data control theory [1]. We will show that our approximation satisfies the original infinite-dimensional constraints provided that the number of discretization is sufficiently large. The finite-dimensional quadratic programming can be easily solved by numerical computation softwares, for example, MATLAB Optimization Toolbox [6]. We also illustrate an example to show the effectiveness of our method.

Notations

In this article, we use the following notations. $\mathbb{R}$, $\mathbb{R}^n$, and $\mathbb{R}^{n\times n}$ are respectively the set of real numbers, n-dimensional real vectors and $n\times n$ matrices. We denote $L^2[0,T]$ by the Lebesgue space consisting of all square integrable real functions on $[0,T]\subset\mathbb{R}$, endowed with the inner product

$$(x,y) := \int_0^T x(t)y(t)\mathrm{d}t, \quad x,y\in L^2[0,T].$$

For a matrix (a vector) M, $M^\top$ is the transpose of M. For a vector $v=[v_1,\dots,v_n]^\top\in\mathbb{R}^n$, $v\preceq 0$ means $v_i\le 0$ for $i=1,2,\dots,n$, and for two vectors v and w, $v\preceq w$ iff $v-w\preceq 0$. For a vector $v\in\mathbb{R}^n$,

$$\|v\|_1 := \sum_{i=1}^n |v_i|.$$

II. Monotone Control Theoretic Splines

We consider the following problem of monotone control theoretic splines:

Problem 1 (Monotone control theoretic splines): Given a linear time-invariant system

$$\dot{x}=Ax+Bu, \quad y=Cx, \quad x(0)=0,$$

where $A\in\mathbb{R}^{n\times n}$, $B\in\mathbb{R}^{n\times 1}$ and $C\in\mathbb{R}^{1\times n}$, and also given a data set $\{(t_1,\alpha_1),(t_2,\alpha_1),\dots(t_N,\alpha_N)\}$, with $0<t_1<t_2<\cdots<t_N=:T$, and $\alpha_1,\dots,\alpha_N\in\mathbb{R}$, find the control $u\in L^2[0,T]$ that minimizes the following cost function:

$$J(u) := \lambda\int_0^T u(t)^2\mathrm{d}t + \sum_{i=1}^N w_i(y(t_i)-\alpha_i)^2, \tag{1}$$

where w_i's are given positive numbers (weights), with the monotonicity constraint

$$\dot{y}(t)\ge 0, \quad \forall t\in[0,T]. \tag{2}$$

This problem has been solved in [2] only when

$$A=\begin{bmatrix}0&1\\0&0\end{bmatrix}, \quad B=\begin{bmatrix}0\\1\end{bmatrix}, \quad C=\begin{bmatrix}1&0\end{bmatrix}.$$

However, this problem has not been solved for other cases.

III. Formulation by semi-infinite quadratic programming

We here formulate our problem by semi-infinite quadratic programming. We first assume that we search the optimal control u in the following subspace of $L^2[0,T]$ as taken in [4]:

$$V_N := \left\{ u \in L^2[0,T] : u = \sum_{i=1}^N \theta_i g_{t_i},\ \theta_i \in \mathbb{R} \right\},$$

where g_t is defined by

$$g_t(\tau) := \begin{cases} Ce^{A(t-\tau)}B, & t > \tau, \\ 0, & t \le \tau. \end{cases}$$

We also denote the inner product in $L^2[0,T]$ by $(\cdot,\cdot)$. By assuming that the control $u \in V_N$, the cost function (1) becomes

$$J\left(\sum_{i=1}^N \theta_i g_{t_i}\right) = \theta^\top (\lambda I + GW) G\theta - 2\alpha^\top GW\theta + \alpha^\top W\alpha,$$

where $\alpha := [\alpha_1, \dots, \alpha_N]^\top$, $\theta := [\theta_1, \dots, \theta_N]^\top$, and

$$G := \begin{bmatrix} (g_{t_1}, g_{t_1}) & (g_{t_2}, g_{t_1}) & \dots & (g_{t_N}, g_{t_1}) \\ \vdots & \vdots & \ddots & \vdots \\ (g_{t_1}, g_{t_N}) & (g_{t_2}, g_{t_N}) & \dots & (g_{t_N}, g_{t_N}) \end{bmatrix},$$
$$W := \mathrm{diag}\{w_1, \dots, w_N\}.$$

Then the derivative $\dot{y}(t)$ is calculated as

$$\begin{aligned} \dot{y}(t) &= C\dot{x}(t) \\ &= CAx(t) + CBu(t) \\ &= CA\int_0^t e^{A(t-\tau)}Bu(\tau)\mathrm{d}\tau + CBu(t) \\ &= CA\int_0^t e^{A(t-\tau)}B\sum_{i=1}^N \theta_i g_{t_i}(\tau)\mathrm{d}\tau + CB\sum_{i=1}^N \theta_i g_{t_i}(t) \\ &= \sum_{i=1}^N \theta_i \left[\int_0^t CAe^{A(t-\tau)}Bg_{t_i}(\tau)\mathrm{d}\tau + CBg_{t_i}(t)\right]. \end{aligned}$$

Let $\dot{g}_t(\tau)$ be the derivative of $g_t(\tau)$, that is,

$$\dot{g}_t(\tau) := \frac{\partial g_t(\tau)}{\partial t} = \begin{cases} CAe^{A(t-\tau)}B, & \text{if } t > \tau \\ 0, & \text{if } t \le \tau, \end{cases}.$$

By using this, $\dot{y}(t)$ can be calculated as

$$\begin{aligned} \dot{y}(t) &= \sum_{i=1}^N \theta_i \left[\int_0^T \dot{g}_t(\tau)g_{t_i}(\tau)\mathrm{d}\tau + CBg_{t_i}(t)\right] \\ &= \sum_{i=1}^N \theta_i \left[(\dot{g}_t, g_{t_i}) + CBg_{t_i}(t)\right] \\ &=: \Phi(t)^\top\theta, \end{aligned}$$

where

$$\Phi(t) := \begin{bmatrix} (\dot{g}_t, g_{t_1}) \\ (\dot{g}_t, g_{t_2}) \\ \vdots \\ (\dot{g}_t, g_{t_N}) \end{bmatrix} + CB\begin{bmatrix} g_{t_1}(t) \\ g_{t_2}(t) \\ \vdots \\ g_{t_N}(t) \end{bmatrix}.$$

The constraint (2) now becomes $\Phi(t)^\top\theta \ge 0$ for all $t \in [0,T]$. In summary, Problem 1 is formulated by the following.

Problem 2: Find $\theta \in \mathbb{R}^N$ that minimize

$$f(\theta) = \frac{1}{2}\theta^\top P\theta + q^\top\theta,$$
$$P := 2(\lambda I + GW)G, \quad q = -2WG\alpha,$$

such that

$$g(\theta, t) = -\Phi(t)^\top\theta \le 0, \quad \text{for all } t \in [0,T]. \tag{3}$$

This is an semi-infinite quadratic programming.

IV. Discretization approach to monotone theoretic control splines

The difficulty of Problem 2 is that the inequality constraint (3) is infinite. We here introduce an approximation technique for such an infinite-dimensional constraint. To reduce this to a finite dimensional one, we adopt the technique of discretization [5], which is also known as fast discretization [1] in sampled-data control theory.

We divide the interval $[0,T]$ into M subintervals:

$$[0,T] = [T_0, T_1] \cup (T_1, T_2] \cup \cdots \cup (T_{M-1}, T_M],$$

where T_i's are discretization points satisfying

$$0 = T_0 < T_1 < \cdots < T_{M-1} < T_M = T.$$

Then we evaluate the function $g(\theta, t)$ in (3) at the discretization points $t = T_0, T_1, \dots, T_M$. To guarantee the constraint (3), we choose real numbers $\epsilon > 0$ and $\lambda > 0$ and adopt the following constraints:

$$\begin{aligned} g(\theta, T_i) &\le -\epsilon, \quad i = 0, 1, 2, \dots, M, \\ -\lambda\mathbf{1} &\preceq \theta \preceq \lambda\mathbf{1}, \end{aligned} \tag{4}$$

where $\mathbf{1} = [1, 1, \dots, 1]^\top \in \mathbb{R}^N$ and $\preceq$ denotes a component wise inequality, that is, the second inequality means $-\lambda \le \theta_i \le \lambda$ for all $i = 1, 2, \dots, N$. Define feasible sets $\mathcal{F}$ and $\mathcal{F}_{\epsilon,M}$ by

$$\mathcal{F} := \left\{\theta \in \mathbb{R}^N : \theta \text{ satisfies (3).}\right\},$$
$$\mathcal{F}_{\epsilon,M} := \left\{\theta \in \mathbb{R}^N : \theta \text{ satisfies (4).}\right\}.$$

Let

$$I_{\max}(M) := \max_{i=1,2,\dots,M} |T_i - T_{i-1}|,$$
$$\mu := \max_{t\in[0,T]} \|\Phi(t)\|_1.$$

Then we have the following proposition:

Proposition 1: Assume $0 < \mu < \infty$. Then $\mathcal{F}_{\epsilon,M} \subseteq \mathcal{F}$ if

$$I_{\max}(M) \le \frac{\epsilon}{\lambda\mu}. \tag{5}$$

Proof. Let $\theta \in \mathcal{F}_{\epsilon,M}$. By (4), θ satisfies

$$g(\theta, T_i) \le -\epsilon, \quad i = 0, 1, \dots, M.$$

Then, for any $t \in [0,T]$, there exists $i \in \{1, \dots, M\}$ such that $t \in [T_{i-1}, T_i]$. Since

$$\max_{t\in[0,T],\ \theta\in\mathcal{F}} \left|\frac{\partial g(\theta,t)}{\partial t}\right| = \max_{t\in[0,T],\ \theta\in\mathcal{F}} \left|\Phi(t)^\top\theta\right| \le \lambda\mu.$$

we have

$$g(\theta,t)-g(\theta,T_i)\leq \lambda\mu(T_i-t).$$

By this inequality, we have

$$\begin{aligned}g(\theta,t)&\leq \lambda\mu(T_i-t)+g(\theta,T_i)\\&\leq \lambda\mu(T_i-t)-\epsilon\\&\leq \lambda\mu I_{\max}(M)-\epsilon\\&\leq 0.\end{aligned}$$

and hence, $\theta\in\mathcal{F}$. Therefore $\mathcal{F}_{\epsilon,M}\subseteq\mathcal{F}$. □

By this proposition, we can guarantee the constraint (3) by searching the parameter θ in the finite-dimensional feasible set $\mathcal{F}_{\epsilon,M}$ provided that the number M is sufficiently large to satisfy (5).

The finite-dimensional constraints (3) can be represented as a matrix inequality,

$$H\theta\preceq v,$$

where

$$H:=\begin{bmatrix}\Phi(T_0)^\top\\ \Phi(T_1)^\top\\ \vdots\\ \Phi(T_M)^\top\\ I\\ -I\end{bmatrix},\quad v:=\begin{bmatrix}-\epsilon\\ -\epsilon\\ \vdots\\ -\epsilon\\ \lambda\mathbf{1}\\ \lambda\mathbf{1}\end{bmatrix}.$$

In summary, the problem of the monotone control theoretic spline for arbitrarily given linear time-invariant system $\{A,B,C\}$ (note that no assumption is needed on A, B and C) can be solved by the following quadratic programming: find $\theta\in\mathbb{R}^N$ which minimizes

$$f(\theta)=\frac{1}{2}\theta^\top P\theta+q^\top\theta,$$

subject to

$$H\theta\preceq v.$$

This is a standard quadratic programming and can be easily solved by MATLAB (using the command `quadprog` in MATLAB Optimization Toolbox [6]).

V. Equality constraints

We can also include equality constraints into our optimization. For example, if we want to have $y(T)=1$ and $\dot{y}(T)=0$, then our constraints are represented by

$$\begin{bmatrix}(g_T,g_{t_1})\\ \vdots\\ (g_T,g_{t_N})\end{bmatrix}^\top\theta=1,\quad \begin{bmatrix}(\dot{g}_T,g_{t_1})\\ \vdots\\ (\dot{g}_T,g_{t_N})\end{bmatrix}^\top\theta=0.$$

These constraints are finite dimensional and can be included easily in our quadratic programming. In general, equality constraints on $y(t)$ and $\dot{y}(t)$, $t\in[0,T]$ can be represented linear constraints as above.

VI. Computing Inner Product

To compute the Grammian G and the matrix $\Phi(t)$, we have to compute the following inner products for fixed $s,\ t\in[0,T]$:

$$(g_s,g_t)=\int_0^T g_s(\tau)g_t(\tau)\mathrm{d}\tau,\quad (\dot{g}_s,g_t)=\int_0^T \dot{g}_s(\tau)g_t(\tau)\mathrm{d}\tau.$$

These values can be easily computed by matrix exponentials [1, Lemma 10.5.1]:

$$\begin{aligned}(g_s,g_t)&=\int_0^T g_s(\tau)g_t(\tau)\mathrm{d}\tau\\&=\int_0^{\min(s,t)}\left[Ce^{A(s-\tau)}B\right]^\top\left[Ce^{A(t-\tau)}B\right]\mathrm{d}\tau\\&=B^\top e^{A^\top s}\left(\int_0^{\min(s,t)}e^{-A^\top\tau}C^\top Ce^{-A\tau}\mathrm{d}\tau\right)e^{At}B\\&=:v_s^\top\left(F_{22}^\top F_{12}\right)v_t,\end{aligned}$$

where $v_\tau:=e^{A\tau}B$ ($\tau=t,s$), and the matrices F_{22} and F_{12} are defined by

$$\begin{bmatrix}F_{11}&F_{12}\\0&F_{22}\end{bmatrix}:=\exp\left(\begin{bmatrix}A^\top&C^\top C\\0&-A\end{bmatrix}h\right),\quad h:=\min(s,t).$$

The inner product $(\dot{g}_s,g_t)$ is also obtained by

$$\begin{gathered}(\dot{g}_s,g_t)=v_s^\top\left(H_{22}^\top H_{12}\right)v_t,\\ \begin{bmatrix}H_{11}&H_{12}\\0&H_{22}\end{bmatrix}:=\exp\left(\begin{bmatrix}A^\top&A^\top C^\top C\\0&-A\end{bmatrix}h\right).\end{gathered}$$

VII. Examples

We here show an example of monotone control theoretic spline. Assume the system is given by

$$A=\begin{bmatrix}0&1&0\\0&0&1\\0&0&0\end{bmatrix},\quad B=\begin{bmatrix}0\\0\\1\end{bmatrix},\quad C=\begin{bmatrix}1&0&0\end{bmatrix}.$$

This system has the transfer function $1/s^3$. Let the original curve be $y_{\text{orig}}(t)=1-e^{-t}$, and the data is obtained by sampling $y_{\text{orig}}(t)$ to which Gaussian noise with $\mathcal{N}(0,0.01)$ is independently added. The estimated parameter is given by

$$\begin{aligned}\theta=[&93.9,\ 185.1,\ 17.1,\ -24.4,\ -7.2,\ 12.1,\\&-10.5,\ 9.7,\ -4.7,\ 0.4,\ 0.3]^\top\end{aligned}\tag{6}$$

The estimation result is shown in Fig. 1. This figure shows that our estimation works well. The derivative of the estimation $y(t)$ is given in Fig. 2. We can see that the derivative of $y(t)$ is always non-negative for all $t\in[0,T]$.

We then consider another system given by

$$A=\begin{bmatrix}1&0\\1&0\end{bmatrix},\quad B=\begin{bmatrix}1\\0\end{bmatrix},\quad C=\begin{bmatrix}0&1\end{bmatrix}.$$

Note that the transfer function of this system is $1/s(s-1)$. For this system, we obtain the optimal θ by the proposed method. The parameter θ is obtained as

$$\begin{aligned}\theta=[&15.05,\ 9.20,\ 0.52,\ -1.70,\ 1.07,\ -0.52,\\&-0.05,\ 0.20,\ 0.24,\ -0.40,\ 0.12,\ ]^\top.\end{aligned}$$

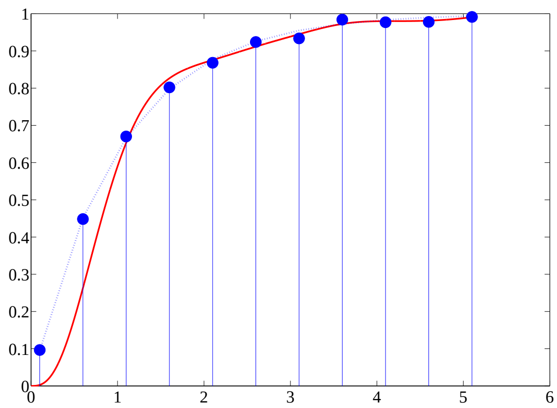

Fig. 1. Data (circles), original curve $y_{\text{orig}}(t) = 1-e^{-t}$ (dash), and estimation $y(t)$ (solid) for the system $1/s^3$

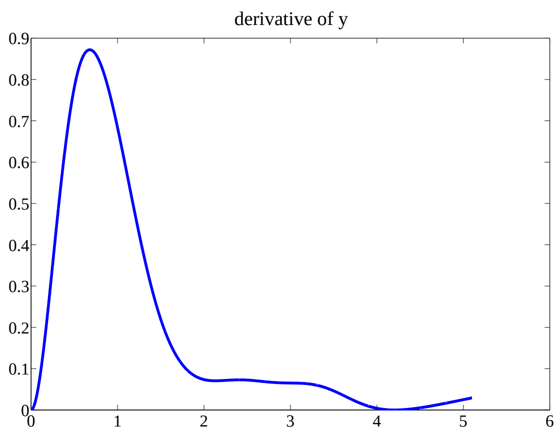

Fig. 2. Derivative of estimation $y(t)$ for the system $1/s^3$

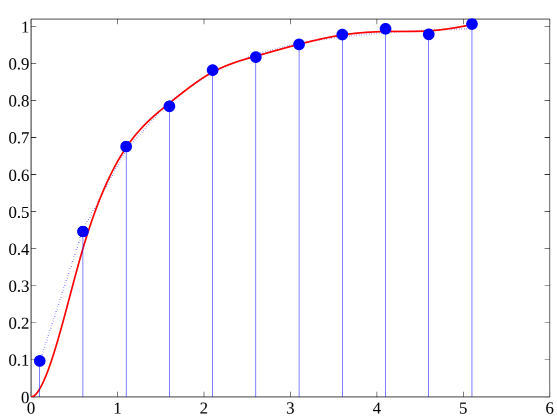

Fig. 3. Data (circles), original curve $y_{\text{orig}}(t) = 1-e^{-t}$ (dash), and estimation $y(t)$ (solid) for the system $1/s(s-1)$

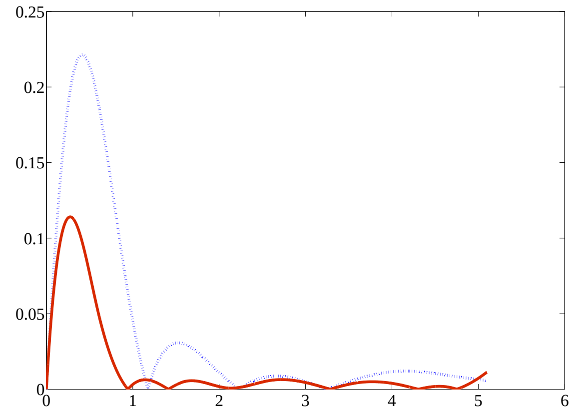

Fig. 4. Absolute estimation errors for $1/s^3$ (dash) and $1/s(s-1)$ (solid)

Note that the values of the elements of this θ is much smaller than that in (6). The estimation result is shown in Fig. 3. We can see that the estimation for the system $1/s(s-1)$ is better than that for $1/s^3$. This is because the system $1/s(s-1)$ has a faster mode $s = 1$. Fig. 4 shows the absolute estimation errors, that is, $|y_{\text{orig}}(t)-y(t)|$. By this figure, the error for $1/s(s-1)$ is much reduced compared with that for $1/s^3$. This example also shows that for given data, the choice of the system $\{A, B, C\}$ considerably affects the accuracy of estimation. How to choose the best $\{A, B, C\}$ for given data is a future work of this study.

VIII. Conclusion

In this article, we have proposed a new method for solving the problem of monotone control theoretic splines for general linear systems. By using fast discretization technique, the problem is described as quadratic programming, and hence the optimal parameters can be easily obtained. Splines with another constraint such as concavity, i.e., $\ddot{y}(t) \geq 0$ can be also solved by the same method.

References

[1] T. Chen and B. Francis, *Optimal Sampled-Data Control Systems*, Springer, 1995.

[2] M. Egerstedt and C.F. Martin, Monotone smoothing splines, *Proceedings of the Mathematical Theory of Networks and Systems*, 2000.

[3] M. Egerstedt and C.F. Martin, *Control Theoretic Splines*, Princeton University Press, 2010.

[4] S. Sun, M. B. Egerstedt, and C. F. Martin, Control theoretic smoothing splines, *IEEE Trans. on Automatics Control*, Vol. 45, No. 12, pp. 2271–2279, 2000.

[5] K. L. Teo and X. Q. Yang, Computational discretization algorithms for functional inequality constrained optimization, *Annals of Operations Research*, Vol. 98, pp. 215–234, 2000.

[6] `http://www.mathworks.com/products/optimization/`.