2008 American Control Conference
Westin Seattle Hotel, Seattle, Washington, USA
June 11-13, 2008

Input Shaping Design Using Sequential Linear Programming (SLP) for Non-Linear Systems

Puneet Singla, Assistant Professor, psingla@buffalo.edu
Tarunraj Singh, Professor, tsingh@buffalo.edu
S.G.Manyam, Graduate Student, sgmanyam@buffalo.edu

Department of Mechanical & Aerospace Engineering
University at Buffalo, Buffalo, NY-14260.

Abstract— **This paper presents a sequential optimization technique for the design of optimal controllers. Linearizing the system model about nominal trajectories results in a linear programming problem which is used to select a perturbation to the nominal control to satisfy the boundary conditions and the state and control constraints. Sequential solution of linear programming problem where the linearized system and control influence matrices are time varying results in the optimal control. Two approaches wherein the perturbation control is constant and linearly varying within a sampling interval are tested on benchmark problems and the results are compared to illustrate the performance of the proposed technique.**

I. Introduction

Shooting method and gradient based iterative approaches have traditionally been used to design optimal controllers. The simplest Input Shapers for linear systems have been derived in closed form [1], however, nonlinear programming is necessary for solving the multi-hump input shaper and switching controllers which minimize maneuver time, fuel consumed etc. [2]–[4]. Recently, Driessen [5] proposed a technique which used linear programming to design Fuel constrained time-optimal control profiles for linear systems, which are close to the globally optimal solution. The discretization of time precludes guaranteeing that a globally optimal solution results. Kim and Singh [6] designed robust controllers for rest-to-rest motion of a vibratory system subject to friction using linear programming. Conord and Singh [7] solved the minimax input shaper for linear systems using LMI which ensures that the globally optimal solution is achieved. Design of Input Shapers for nonlinear systems have required nonlinear programming [8], [9] which are sensitive to initial guesses. There is clearly a need for a technique which can be used to design input shapers for nonlinear systems without assuming any structure for the control profile.

In this paper an iterative approach using linear programming for solving the optimal control profiles is discussed. The nonlinear dynamic system is linearized over time with a finite number of time intervals. The final states of the system are expressed as a function of control values and states at each discretized point of time. This is posed as a linear programming problem with perturbations to the control values at each interval of time as the unknowns to be solved for. The problem is solved iteratively till the terminal constraints are satisfied. An outer loop which optimizes for the cost which corresponds to a terminal state or maneuver time is selected using a bisection algorithm. Time-optimal control problem is a special case of this where the cost would be final time. Integral cost function can be converted to terminal state cost by augmenting the state space model. Solving for the optimal final value of that particular state gives the control profile for the optimal nonlinear integral cost. In the next section the algorithm for solving optimal control profile is discussed. This involved linearizing the nonlinear system about a nominal trajectory and subsequently solving a linear programming problem in which the perturbation to the nominal control is assumed to be constant over a sampling interval. A technique to convert free final time problem to a fixed final time problem is also described which has shown to result in faster convergence. This entails introducing an additional variables into the state space which needs to be optimized for. An extension to the aforementioned technique where the control is assumed to vary linearly over a sampling interval is also presented. The proposed technique is illustrated on two benchmark nonlinear control problems.

II. Optimal Control Problem

A general optimal control problem can be stated as: *given a model of system dynamics and the constraints on state and control variables, compute the appropriate control vector that will drive the system to the desired*

978-1-4244-2079-7/08/$25.00 ©2008 AACC.

state trajectory while minimizing a performance index

$$\min_{\mathbf{u}(t),t_f} \quad J = \int_0^{t_f} g(\bar{\mathbf{x}}(t), \mathbf{u}(t))dt, \; t_f \text{ is free} \quad (1)$$

subject to

$$\dot{\bar{\mathbf{x}}} = \bar{f}(\bar{\mathbf{x}}, \mathbf{u}), \; \bar{\mathbf{x}}(0) = \bar{\mathbf{x}}_0, \; F(\bar{\mathbf{x}}(t_f)) = 0 \quad (2)$$

$$\mathcal{N}\bar{\mathbf{x}}(t) \leq N_0, \; \mathbf{u}_l \leq \mathbf{u} \leq \mathbf{u}_u, \quad \forall t \quad (3)$$

where, $\bar{\mathbf{x}} \in \mathbb{R}^n$ in the state vector and $\mathbf{u} \in \mathbb{R}^m$ is the control vector. The aforementioned optimal control problem can be converted to a Mayer problem by defining a new state vector $\mathbf{x} \triangleq \left\{ \bar{\mathbf{x}}, \int\limits_0^t g(\bar{\mathbf{x}}(t), \mathbf{u}(t))dt \right\}^T$ and our new objective is to minimize the final value of $\mathbf{x}_{n+1}(t_f) = J$. Hence, the modified optimal control problem can be stated as

$$\min_{\mathbf{u}(t),t_f} \quad \mathbf{x}_{n+1}(t_f), \; t_f \text{ is free}$$

subject to

$$\dot{\mathbf{x}} = \left\{ \begin{array}{c} \bar{f}(\bar{\mathbf{x}}, \mathbf{u}) \\ g(\bar{\mathbf{x}}(t), \mathbf{u}(t)) \end{array} \right\} = f(\mathbf{x}, \mathbf{u}), \; F(\mathbf{x}(t_f)) = 0$$

$$\mathbf{x}(0) = \left[\begin{array}{cccc} \mathbf{x}_1(0) & \cdots & \mathbf{x}_n(0) & 0 \end{array} \right]^T = \mathbf{x}_0$$

$$\mathcal{N}\mathbf{x}(t) \leq N_0, \; \mathbf{u}_l \leq \mathbf{u} \leq \mathbf{u}_u, \quad \forall t$$

Further, by defining the normalizing time, $\tau \triangleq \frac{t}{t_f}$ and including t_f as one of the parameter to be optimized, we can rewrite the optimal control control problem as:

$$\min_{\mathbf{u}(\tau),t_f} \quad \mathbf{x}_{n+1}(1) \quad (4)$$

subject to

$$\mathbf{x}' = t_f f(\mathbf{x}, \mathbf{u}) = h(\mathbf{x}, \mathbf{u}, t_f), \; (.)' = \frac{d}{d\tau} \quad (5)$$

$$\mathbf{x}(0) = \mathbf{x}_0, \; F(\mathbf{x}(1)) = 0 \quad (6)$$

$$\mathcal{N}\mathbf{x}(\tau) \leq N_0, \; \mathbf{u}_l \leq \mathbf{u} \leq \mathbf{u}_u, \quad \forall \tau \quad (7)$$

The main advantage of formulating the optimal control problem as Mayer's problem is that performance index for Mayer's problem is linear in nature and thus, one can make use of SLP algorithms to solve the problem as discussed in our earlier work for time optimal control [10].

A. SLP for Optimal Control of Nonlinear Systems

Let us consider the optimal control problem of Eqs. (4)-(7). Like any nonlinear optimal control method, we desire to compute the optimal control iteratively by assuming an initial control profile $\mathbf{u}_0(t)$ and determining the corresponding evolution of the states. To determine the update to the control profile, we need a mechanism which exploits the error in terminal conditions to perturb the current control profile. We make use of the SLP framework, introduced for the time optimal control problem in Ref. [10], to find the feasible perturbations to an initial control problem.

Approximating the nonlinear system as a series of linear systems obtained by linearizing the nonlinear model at discrete time intervals $t_k, \; k = 1, 2, \cdots, N$, the system dynamics can be written as

$$\dot{\mathbf{x}}(t_k) + \Delta\dot{\mathbf{x}} = h(\mathbf{x}(t_k), \mathbf{u}, t_f) + \frac{\partial h}{\partial \mathbf{x}}|_{\mathbf{x}(t_k)}\Delta\mathbf{x}$$
$$+ \frac{\partial h}{\partial \mathbf{u}}|_{\mathbf{u}(t_k)}\Delta\mathbf{u} + \frac{\partial h}{\partial t_f}\Delta t_f, \; t_k \leq t \leq t_{k+1} \quad (8)$$

which can be simplified to

$$\Delta\dot{\mathbf{x}} = \underbrace{\frac{\partial h}{\partial \mathbf{x}}|_{\mathbf{x}(t_k)}}_{\mathbf{A}_k}\Delta\mathbf{x} + \underbrace{\frac{\partial h}{\partial \mathbf{u}}|_{\mathbf{u}(t_k)}}_{\mathbf{B}_k}\Delta\mathbf{u} + \underbrace{\frac{\partial h}{\partial t_f}}_{\mathbf{C}_k}\Delta t_f \quad (9)$$

The closed form solution of the system can be written as:

$$\Delta\mathbf{x}(t) = \exp\left(\mathbf{A}_k(t - t_k)\right)\mathbf{x}(t) + \int\limits_{t_k}^{t} \exp\left(\mathbf{A}_k(t - \tau)\right)$$
$$+ \left[\mathbf{B}_k\Delta\mathbf{u}(\tau) + \mathbf{C}_k\Delta t_f\right]d\tau, \; t_k \leq t \leq t_{k+1}$$

An analytical expression for $\Delta\mathbf{x}(t)$ can be obtained by appending the perturbed state vector $\Delta\mathbf{x}$ with perturbed final time, Δt_f, perturbed control vector $\Delta\mathbf{u}$ and its higher derivatives depending upon the assumed time profile for $\Delta\mathbf{u}$. For example, if we approximate perturbed control profile $\Delta\mathbf{u}(t)$ as a piecewise constant function, then the augmented system can be represented as:

$$\left\{ \begin{array}{c} \Delta\dot{\mathbf{x}} \\ \Delta\dot{\mathbf{u}} \\ \Delta\dot{\mathbf{t}}_f \end{array} \right\} = \mathcal{A}_k \left\{ \begin{array}{c} \Delta\mathbf{x} \\ \Delta\mathbf{u} \\ \Delta\mathbf{t}_f \end{array} \right\}, \; t_k \leq t \leq t_{k+1} \quad (11)$$

where,

$$\mathcal{A}_k = \left[\begin{array}{ccc} \mathbf{A}_k & \mathbf{B}_k & \mathbf{C}_k \\ \mathbf{O}_{(m+1)\times n} & \mathbf{O}_{(m+1)\times m} & \mathbf{O}_{(m+1)\times 1} \end{array} \right]$$

whose solution is given by

$$\left\{ \begin{array}{c} \Delta\mathbf{x}(t) \\ \Delta\mathbf{u}(t) \\ \Delta\mathbf{t}_f \end{array} \right\} = \underbrace{\exp\left(\mathcal{A}_k(t - t_k)\right)}_{\Phi(t,t_k)} \left\{ \begin{array}{c} \Delta\mathbf{x}(t_k) \\ \Delta\mathbf{u}(t_k) \\ \Delta\mathbf{t}_f \end{array} \right\} \quad (12)$$

and the state transition matrix $\Phi(t, t_k)$ can be partitioned as:

$$\Phi(t, t_k) = \left[\begin{array}{ccc} \mathbf{G}_k & \mathbf{H}_k & \mathbf{L}_k \\ \mathbf{O}_{m\times n} & \mathbf{I}_{m\times m} & \mathbf{O}_{m\times 1} \\ \mathbf{O}_{1\times n} & \mathbf{O}_{1\times m} & 1 \end{array} \right] \quad (13)$$

Thus, the state response for the piecewise constant control input $\Delta\mathbf{u}(k)$ can be written as:

$$\Delta\mathbf{x}(k+1) = \mathbf{G}_k\Delta\mathbf{x}(k) + \mathbf{H}_k\Delta\mathbf{u}(k) + \mathbf{L}_k\Delta t_f \quad (14)$$

which can further be simplified to

$$\Delta\mathbf{x}(k+1) = \left(\prod_{i=1}^{k}\mathbf{G}_i\right)\Delta\mathbf{x}(1) + \mathbf{H}_k\Delta\mathbf{u}(k) + \sum_{i=1}^{k-1}\left(\prod_{j=i+1}^{k}\mathbf{G}_j\right)[\mathbf{H}_i\Delta\mathbf{u}(i) + \mathbf{L}_i\Delta t_f] + \mathbf{L}_k\Delta t_f$$

where $\Delta\mathbf{x}(1)$ represents the initial perturbation state of the system and is zero, since the initial condition are prescribed. To solve the control problem with specified initial and final states, in addition to the final time (t_f), the final state constraint can be represented as

$$\Delta\mathbf{x}(N+1) = \mathbf{H}_N\Delta\mathbf{u}(N) + \mathbf{L}_N\Delta t_f + \sum_{i=1}^{N-1}\left(\prod_{j=i+1}^{N}\mathbf{G}_j\right)[\mathbf{H}_i\Delta\mathbf{u}(i) + \mathbf{L}_i\Delta t_f] \quad (16a)$$

Similarly, to determine the response of the perturbed system of Eq. (9) to a piecewise linear input with a slope of $\frac{\mathbf{u}_{k+1}-\mathbf{u}_k}{t_{k+1}-t_k}$ in time interval $[t_k,\ t_{k+1}]$, we can augment the perturbed dynamical system with following equations:

$$\Delta\ddot{\mathbf{u}} = 0,\ \Delta\dot{t}_f = 0 \quad (17)$$

resulting in the following augmented system:

$$\left\{\begin{array}{c}\Delta\dot{\mathbf{x}}\\ \Delta\dot{\mathbf{u}}\\ \Delta\ddot{\mathbf{u}}\\ \Delta\dot{\mathbf{t}}_f\end{array}\right\} = \mathcal{A}_k\left\{\begin{array}{c}\Delta\mathbf{x}\\ \Delta\mathbf{u}\\ \Delta\dot{\mathbf{u}}\\ \Delta\mathbf{t}_f\end{array}\right\},\ t_k \le t \le t_{k+1} \quad (18)$$

where,

$$\mathcal{A}_k = \begin{bmatrix}\mathbf{A}_k & \mathbf{B}_k & \mathbf{O}_{n\times m} & \mathbf{C}_k\\ \mathbf{O}_{m\times n} & \mathbf{O}_{m\times m} & \mathbf{I}_{m\times m} & \mathbf{O}_{m\times 1}\\ \mathbf{O}_{m\times n} & \mathbf{O}_{m\times m} & \mathbf{O}_{m\times m} & \mathbf{O}_{m\times 1}\\ \mathbf{O}_{1\times n} & \mathbf{O}_{1\times m} & \mathbf{O}_{1\times m} & \mathbf{O}_{1\times 1}\end{bmatrix}$$

Once again, the state transition matrix, $\Phi(t,t_k) = \exp(\mathcal{A}(t-t_k))$ can be partitioned as:

$$\Phi(t,t_k) = \begin{bmatrix}\mathbf{G}_k & \mathbf{\Gamma}_k & \mathbf{\Psi}_k & \mathbf{L}_k\\ \mathbf{O}_{m\times n} & \mathbf{I}_{m\times m} & T\mathbf{I}_{m\times m} & \mathbf{O}_{m\times 1}\\ \mathbf{O}_{m\times n} & \mathbf{O}_{m\times m} & \mathbf{I}_{m\times m} & \mathbf{O}_{m\times 1}\\ \mathbf{O}_{1\times n} & \mathbf{O}_{1\times m} & \mathbf{O}_{1\times m} & 1\end{bmatrix}$$

where T is the sampling time. Now, the solution of the perturbed linear system of Eq. (18) can be written as:

$$\Delta\mathbf{x}(k+1) = \mathbf{G}_k\Delta\mathbf{x}(k) + \mathbf{\Xi}_k\Delta\mathbf{u}(k) + \mathbf{\Theta}_k\Delta\mathbf{u}(k+1) + \mathbf{L}_k\Delta t_f \quad (19a)$$

where

$$\mathbf{\Xi}_k = \mathbf{\Gamma}_k - \frac{\mathbf{\Psi}_k}{T},\ \mathbf{\Theta}_k = \frac{\mathbf{\Psi}_k}{T}$$

Further, simplification leads to

$$\Delta\mathbf{x}(k+1) = \left(\prod_{i=1}^{k}\mathbf{G}_i\right)\Delta\mathbf{x}(1) + \mathbf{\Xi}_k\Delta\mathbf{u}(k) + \sum_{i=1}^{k-1}\left(\prod_{j=i+1}^{k}\mathbf{G}_j\right)[\mathbf{\Xi}_i\Delta\mathbf{u}(i) + \mathbf{\Theta}_i\Delta\mathbf{u}(i+1) + \mathbf{L}_i\Delta t_f] + \mathbf{\Theta}_k\Delta\mathbf{u}(k+1) + \mathbf{L}_k\Delta t_f \quad (21a)$$

$\Delta\mathbf{x}(1)$ is the initial perturbation state of the system, and it is equal to zero since we know the initial states of the system. The final state of the system can be represented as:

$$\Delta\mathbf{x}(N+1) = \left(\prod_{i=1}^{N}\mathbf{G}_i\right)\Delta\mathbf{x}(1) + \mathbf{\Xi}_N\Delta\mathbf{u}(k) + \sum_{i=1}^{N-1}\left(\prod_{j=i+1}^{N}\mathbf{G}_j\right)[\mathbf{\Xi}_i\Delta\mathbf{u}(i) + \mathbf{\Theta}_i\Delta\mathbf{u}(i+1) + \mathbf{L}_i\Delta t_f] + \mathbf{\Theta}_N\Delta\mathbf{u}(N+1) + \mathbf{L}_N\Delta t_f \quad (22a)$$

In summary, if we approximate perturbed control profile $\Delta\mathbf{u}(t)$ as a piecewise polynomial function, then, the response to the perturbed system of Eq. (9) can be written as a linear function of perturbed control $\Delta\mathbf{u}(t)$ and perturbed time Δt_f:

$$\Delta\mathbf{x}(k+1) = \Upsilon_k\Delta\mathbf{u}_{1k} + \Pi_k\Delta t_f,\ k = 1,2,\cdots,N \quad (23)$$

where,

$$\Delta\mathbf{u}_{1k} = \{\Delta\mathbf{u}_1,\ \Delta\mathbf{u}_2,\ \cdots,\ \Delta\mathbf{u}_k\}^T$$

Thus, the constraints on state vector $\mathbf{x}(t)$ can be written as:

$$\Delta\mathcal{N}_k = N_0 - \mathcal{N}\mathbf{x}_k \ge \mathcal{N}\Delta\mathbf{x}_k = \underbrace{\mathcal{N}\begin{bmatrix}\Upsilon_k & \Pi_k\end{bmatrix}}_{\Omega_k}\left\{\begin{array}{c}\Delta\mathbf{u}_{1k}\\ \Delta t_f\end{array}\right\}$$

Similarly, the terminal state constraints can be written as:

$$\Delta F_N = 0 - F(\mathbf{x}_N) = \frac{\partial F}{\partial\mathbf{x}}|_{\mathbf{x}_N}\Delta\mathbf{x}_N = \underbrace{\frac{\partial F}{\partial\mathbf{x}}|_{\mathbf{x}_N}\begin{bmatrix}\Upsilon_N & \Pi_N\end{bmatrix}}_{\Omega_N}\left\{\begin{array}{c}\Delta\mathbf{u}_{1N}\\ \Delta t_f\end{array}\right\} \quad (25a)$$

ΔN_k and ΔF_N are the difference between the desired state constraints and the state constraints resulting from the nominal control $\mathbf{u}(t)$. Since the final value of the state corresponding to performance index J, the determination of the optimal control profile requires that the initial estimate of the performance index J should be used to determine the feasibility of satisfying all the constraints. Now, we propose an algorithm which approximate the solution to the original nonlinear optimal control problem posed by Eqs. (4)-(7) by solving the following linear programming problem recursively.

$$\min_{\Delta \mathbf{u}_{1N}, \Delta t_f} \underline{a}^T \{\Delta \mathbf{u}_{1N} \ \Delta t_f\}^T \tag{26}$$

subject to

$$\begin{bmatrix} \Delta N_k & \Delta F_N \end{bmatrix} = \begin{bmatrix} \Omega_k & \Omega_N \end{bmatrix} \left\{ \begin{array}{c} \Delta \mathbf{u}_{1N} \\ \Delta t_f \end{array} \right\}, \tag{27}$$

$$\mathbf{x}_n(1) = J_f, \ \mathbf{u}_l - \mathbf{u} \leq \Delta \mathbf{u}_{1N} \leq \mathbf{u}_u - \mathbf{u} \tag{28}$$

We get a feasible solution for linearized system dynamics by solving the aforementioned LP problem at each iteration which differs from the true nonlinear state constraints. We anticipate that at each iteration the linearization error decreases and finally, we will obtain the solution to the original optimal control problem. For a generic optimal control problem, the main steps of such an algorithm are enumerated as:

1) Guess the bounds for performance index, J_f^L and J_f^U.
2) Initialize $J_f = \frac{J_f^L + J_f^U}{2}$ and divide the time interval $[0 - t_f]$ into pre-specified N intervals and guess the value for control variable $\mathbf{u}(i), \quad i \in [1, N]$ compatible with actuator constraints.
3) Integrate the nonlinear system dynamics Eq. (5), to compute $\mathbf{x}(1)$ and if intermediate time and terminal state constraints are satisfied then decrease the value of upper bound on the performance index to the current guess for the performance index and *Go to Step 2*.
4) Else linearize the nonlinear dynamics system and find a feasible solution by solving the LP problem posed by Eqs. (26)-(28).
5) If the solution to the LP problem (Eqs. (26)-(28)) exists, then modify the initial guess for control $\mathbf{u}_{new}(i) = \mathbf{u}_{old}(i) + \alpha \Delta \mathbf{u}_{1N}(i), \ 0 \leq \alpha \leq 1$ and *Go To Step 3*.
6) Else, increase the value of lower bound on the performance index to the current guess for the performance index and *Go To Step* 2.

Finally, it should be noticed that with the proposed algorithm one can always impose system dynamics constraints using continuous differential equations without any approximation, while other nonlinear programming algorithms [11] require the discretization of the system dynamics and constraints to be written as algebraic equations. Hence, one needs to approximate the continuous time differential equations with discrete time difference equations and as a consequence of this, the optimal solution is always accurate up to the errors introduced by the discretization process.

III. Numerical Examples

To illustrate the proposed technique, we consider the following two benchmark problems.

A. Goddard Rocket

This problem was first posed by R. H. Goddard in 1919 when he was building a rocket to be fired vertically to reach high altitudes. Later this was studied extensively by various people [12]–[14]. The problem is to find the thrust history to maximize the final altitude of a vertically launched rocket.

In this paper the following assumptions were made to simplify the problem as listed in Refs. [15]. The rocket is regarded as a point variable mass, flying over a flat stationary Earth with Newtonian central gravitational field. Let h is the altitude of the rocket, v be its velocity and m is the varying mass of the rocket. Then, the equations of motion for the rocket are given as:

$$\dot{h} = v, \ m\dot{v} = T - D(h, v) - mg(h), \ \dot{m} = -\frac{T}{c}$$

where

$$D(h, v) = D_c v^2 exp\left(-h_c \left(\frac{h - h(0)}{h(0)}\right)\right)$$

$$g(h) = g_0 \left(\frac{h(0)}{h}\right)^2$$

g_0 is the gravitational force at the earth's surface.The final mass is constrained to be a fraction of initial mass, $m_c m(0)$. The constraints on the states and control are:

$$m(t_f) \leq m(t) \leq m(0), \quad h(t) \geq h(0)$$
$$v(t) \geq 0, \ 0 \leq T \leq T_{max}$$

where,

$$T_{max} = 3.5 g_0 m(0), D_c = \frac{1}{2} v_c \frac{m(0)}{g_0}, c = \frac{1}{2} (g_0 h(0))^{\frac{1}{2}}$$
$$h(0) = m(0) = g_0 = 1, h_c = 500, m_c = 0.6, v_c = 620$$

Fig. 1 shows a three dimensional plot illustrating the feasibility of the problem against time for different heights. The SLP algorithm is used to solve this problem as discussed in the previous section while approximating optimal control profile as both piecewise constant and linear function. Fig. 2(a), shows the time profile of the

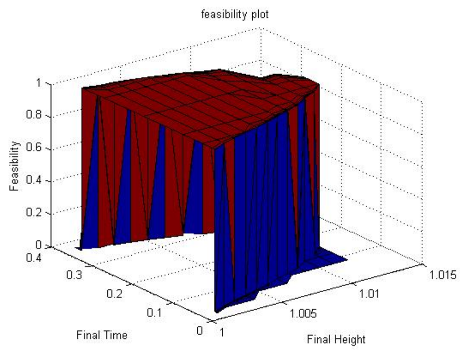


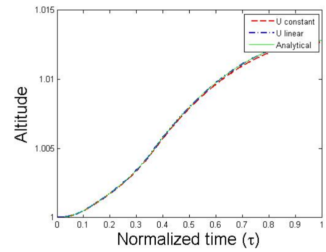
Fig. 1. Feasibility against time for different heights of the rocket

TABLE I

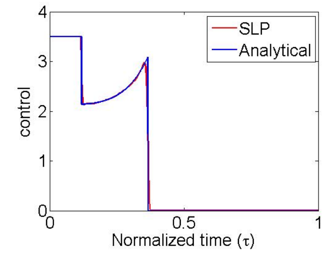
RESULTS COMPARISON FOR GODDARD ROCKET PROBLEM

Algorithm	Maximum Altitude (h_f)	Constr Violation
SNOPT	1.01282	0.011
Proposed Algorithm	1.0128369	$1e^{-4}$

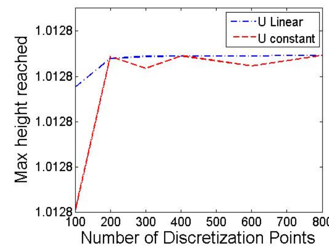
rocket altitude. The solid line corresponds to piecewise constant assumption for optimal control profile while dotted line corresponds to piecewise linear approximation for optimal control profile. Fig. 2(b) shows the plots for computed and analytical thrust profile as a function of normalized time. Further, Fig. 2(c) shows the maximum rocket altitude, for both with piecewise constant and linear approximation on optimal control profile, as a function of number of discretization steps. From these plots, it is clear that the optimal solution converges well with reasonable number of discretization steps $(N = 400)$ and as expected, the piecewise linear approximation solution is more accurate than the solution based upon the piecewise constant assumption. Finally, Table I shows the comparison of the proposed algorithm with SNOPT [15]. It should be mentioned that with the proposed algorithm one can impose system dynamics constraints in a continuous manner using continuous differential equations, while SNOPT requires the constraints to be written as algebraic equations. Hence, one needs to approximate the continuous time differential equations with discrete time difference equations and as a consequence of this, the optimal solution is always accurate up to the errors introduced by the discretization process. The optimal control profile from SNOPT is integrated with the state equations to determine the constraint violation which is presented in the last column



(a) Altitude



(b) Thrust



(c) Height Vs Discretization points

Fig. 2. Simulation Results for the Goddard Problem

of the table. It is clear that the proposed algorithm converges to a better solution in terms of the optimal cost, constraint violations and the number of optimization variables.

B. Thrusted Skate Problem

Thrusted Skate is a class of problem where one actuator provides the thrust and the other one controls the steering angle. The dynamics of the system is [16]:

$$\dot{x} = vcos(\theta), \;\; \dot{y} = vsin(\theta), \;\; \dot{v} = usin(\theta) \qquad (30a)$$

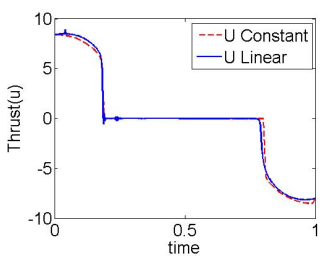
The goal is to find the optimal profiles for v and θ which minimizes a performance index $J = \int_0^1 u(t)^2 dt$ while moving the skate from one point to another point. To find the optimum cost, the performance index J is added as one of the states

$$\dot{j} = u^2, \quad j(0) = 0; \qquad (31)$$

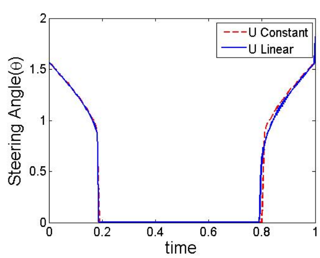
The final value of j is minimized using the SLP algorithm as discussed in the paper. The initial and final values of the state variables are given as:

$$x(0) = y(0) = v(0) = j(0) = 0$$
$$x(1) = \cos(\pi/12),\ y(1) = \sin(\pi/12),\ v(1) = 0$$

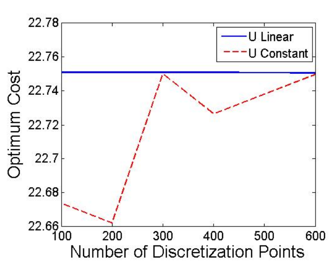
The SLP algorithm is used to solve this problem as discussed in the previous section while approximating the optimal control profile as both piecewise constant and linear function. Figs. 3(a) and 3(b) show the time profile for skate thrust u and steering angle θ, respectively. Further, Fig. 3(c) shows the optimal value for the performance index J as a function of number of discretization steps for the piecewise constant and linear approximation of the optimal control profile. From these plots, it is clear that the optimal solution converges well with reasonable number of discretization steps ($N = 400$) and as expected, the piecewise linear approximation solution is more accurate than the solution based upon the piecewise constant assumption.



(a) Thrust (u)



(b) Steering Angle(θ)



(c) Optimum Cost Vs Number of time steps

Fig. 3. Simulation Results for the Thrusted skate problem

IV. Conclusions

This paper proposed a SLP approach for the determination of time-optimal controllers for systems with nonlinear dynamics. Evaluating its performance on benchmark problems, it is clear that it outperforms standard nonlinear programming solvers as delineated by Dolan et al. (COPS). The obvious benefit of this approach is that no prior knowledge of the structure of the control profile is necessary to initiate the algorithm. Finally, the preliminary results presented here provide compelling evidence for the merits of the proposed approach. The authors are currently extending this technique to have higher order hold approximations for control variable between two time steps.

References

[1] N. Singer and W. P. Seering. Pre-shaping command inputs to reduce system vibration. 112(1):76–82, 1990.

[2] S. Derezinski W. Singhose and N. Singer. Extra-insensitive input shapers for controlling flexible spacecraft. 19(2):385–391, 1996.

[3] T. Singh and S. R. Vadali. Robust time-optimal control: A frequency domain approach. *Journal of Guidance, Control and Dynamics*, 17(2):346–353, Mar. 1994.

[4] T. Singh W. Singhose and W. Seering. On-off control with specified fuel usage. *ASME Journal of Dynamic Systems, Measurement and Control*, 121(2):206–212, Apr. 1999.

[5] B. J. Driessen. On-off minimum-time control with limited fuel usage: Near global optima via linear programming. *Optimal Control Applications and Methods*, 27(3):161–168, 2006.

[6] J-J. Kim and T. Singh. Desensitized control of vibratory systems with friction: linear programming approach. *Optimal Control: Applications and Methods*, 25(4):165–180, July 2004.

[7] T. Conord and T. Singh. Robust input shaper design using linear matrix inequalities.

[8] A. E. Bryson and R. A. Mills. Linear-quadratic-gaussian controllers with specified parameter robustness. 21(1):11–18, February 1998.

[9] T. Singh S.W. Liu. Robust time-optimal control of nonlinear structures with parameter uncertainties. *ASME Journal of Dynamic Systems, Measurement and Control*, 119(4):743–748, Dec. 1997.

[10] T. Singh and P. Singla. Sequential linear programming for design of time-optimal controllers. In 46^{th} *IEEE Conference on Decision and Control (*CDC*)*, 2007.

[11] P. E. Gill, W. Murray, and M. A. Saunders. SNOPT: An SQP algorithm for large-scale constrained optimization. *j-SIAM-J-OPT*, 12(4):979–1006, 2002.

[12] B. Garifinkel. A solution of the goddard problem. *Journal of SIAM Control*, 1(3):349–368, 1963.

[13] H. Munick. Goddard problem with bounded thrust. *American Institute of Aeronautics and Astronautics Journal*, 3(7):1283–1285, 1965.

[14] H. S. Tsien and R. C. Evans. Optimum thrust programming for a sounding rocket. *Journal of the American Rocket Society*, 21(5):99–107, 1951.

[15] E. Dolan, J. J. Moré, and T. S. Munson. Benchmarking optimization software with cops 3.0. Technical Report ANL/MCS-273, Mathematics and Computer Science Division, Argonne National Laboratory, Argonne, IL, 2004.

[16] K. M. Lynch. Optimal control of the thrusted skate. *Automatica*, 39:173 – 176, 2003.