CHAPTER 3

A convex optimization approach for solving time optimal control problems

Let us recall that, by considering the necessary condition for optimality of the solution for the linear time-optimal control problem, it is seen that the bang-bang solution (2.34) is uniquely determined by the co-state variable $\lambda(t)$ whose trajectory is defined by (2.31). The complete solution of can be easily computed as

$$\lambda(t) = e^{-A^T t}\lambda_0 \tag{3.1}$$

which requires knowledge of the initial co-state λ_0. Thus $u^*(t)$ is uniquely determined from λ_0. However, in previous researches, the solutions for $u(t)$ are computed beforehand by assuming a bang-bang form and solving for switching times. This has been done both numerically and analytically (for linear cases). The solution is then checked afterwards for optimality by trying to solve (2.37), rather than using (3.1) directly to find the optimal solution.

In this chapter a mathematical algorithm for finding time-optimal control input by searching over possible initial values for the co-state vector is proposed. The convergence of the algorithm depends on the geometric properties of the reachable set, which will be described in section 3.1, following the approach of Hermes and Lasalle [34]. In section 3.2 the iterative scheme is introduced followed by the explanation and remarks for each steps. The numerical examples of the linear cases are given in section 3.3. Later in this chapter, the proposed methods is then shown to be capable of solving adapted problems for system models with non-linear Coulomb friction. The solutions are then used for simulation with selected example cases of flexible structures.

3.1 The reachable set and an illustrative example

The general solution of (2.1) is

$$x(t) = X^{-1}(t)[x_0 + \int_0^t X(\tau)Bu(\tau)d\tau] \tag{3.2}$$

where x_0 is the initial state vectors and, $X(t)$ is the inverse (backwards) state transition matrix. When A is constant, $X(t) = e^{-At}$ and $X(0) = I$ (identity matrix).

Let $\mathcal{A}(t)$ be the set of all possible $x(t)$ that can be reached at time t using all possible $u(t) \in \mathcal{U}$, then $\mathcal{A}(t)$ is called the attainable set at time t. Next, define

$$Y(t) = X(t)B \tag{3.3}$$

and

$$\dot{y}(t) = Y(t)u(t). \tag{3.4}$$

The reachable set $\mathcal{R}(t)$ is the set of all possible $y(t)$ that can be reached within time t. The relation between $x(t)$ and $y(t)$ is that state $x(t)$ reaching final state $x_f \in \mathcal{A}(t)$ is equivalent to $y(t)$ reaching $w(t) \in \mathcal{R}(t)$ where $w(t) = X(t)x_f - x_0$.

From the definition of $y(t)$ in (3.3) and (3.4), it can be proved that $\mathcal{R}(t)$ has the following properties [34]

1. $\mathcal{R}(t)$ is convex, closed, symmetric and contains the origin.
2. $\mathcal{R}(t)$ is continuously expanding so that $\mathcal{R}(t_1) \subset \mathcal{R}(t_2),\ t_2 > t_1 > 0$.
3. A point $w^* = y(t^*; u^*)$ in the boundary of the reachable set is reached by a unique optimal control u^*, and t^* is the minimum time for transfer.

From this set of properties, an alternate way to define the objective of the optimal control problem is for $y(t)$ to hit the target point $w(t)$ in minimum time. Equivalently, this corresponds to the situation where t is increased from zero so that $\mathcal{R}(t)$ expands until the boundary of $\mathcal{R}(t)$ first contains the point $w(t)$. Thus, $u^*(t)$ is optimal if and only if $w(t^*) \in \mathcal{R}(t^*)$ and $w(t) \notin \mathcal{R}(t)$ for $t < t^*$; then t^* is the minimum time.

The optimal control u^* also has the property that it maximizes the rate of change of y (as defined by (3.4)) in some fixed direction η, where η is a nonzero vector in $\mathbb{R}^n$. Thus, u^* maximizes the value

$$\eta^T \dot{y} = \eta^T Y(t)u(t)$$

at all time and also has the bang-bang form

$$u^*(t) = \text{sgn}\left[\eta^T Y(t)\right]. \tag{3.5}$$

Note that, for multi-input systems, equation (3.5) means that for each $i = 1, 2, ..., m$ we have $u_i^*(t) = \text{sgn}\left[\eta^T Y(t)\right]_i$ when $\left[\eta^T Y(t)\right]_i \neq 0$.

It follows from (2.31), (2.34), (3.4) and (3.5), that η must be equivalent to the initial value of the co-state vector λ_0. Since η maximizes the rate of change of y at any point w^* in the boundary (surface) of $\mathcal{R}(t^*)$, then there will be a supporting hyperplane $\pi(\eta)$ passing through w^* with η as an outward normal direction of this plane. Therefore for any fixed t^*, any point w^* can be reached by an optimal control input u^*. The calculation of w^* may be performed by integrating (3.4) subject to (3.5). This could be written as a single-valued mapping from η to the point w^* as

$$w^* = f\left(\eta, t^*\right). \tag{3.6}$$

The mapping (3.6) implies that w^* is uniquely determined by, and can be directly calculated from, η.

Suppose the *i*th component of the optimal input u^* involves l^i switches at times $t_1^i, t_2^i, ..., t_l^i$. According to (3.5), at these instances the value of sgn $\left[\left(\eta^T X(t_j)B\right)_i\right]$ must switch between -1 and 1. Thus, a further property of the optimal solution is that

$$\eta^T V_i = [0\ 0\ \cdots\ 0]. \tag{3.7}$$

where $V_i = \left[X(t_1^i)B \quad X(t_2^i)B \quad \cdots \quad X(t_{l^i}^i)B\right]$.

3.1.1 Illustrative example

Let us consider the three-states linear system representing motion of a flexible structure whose states x_1, x_2 and x_3 correspond to an overall motion and the vibratory states respectively. The characteristic parameters given are $\omega_n = 1$ rad/sec and $\zeta = 0.03$ with the constraint on the control input following (2.22). The state-space model of the system following (2.1) and (2.3) is

$$\dot{x}(t) = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & 1 \\ 0 & -1 & -0.06 \end{bmatrix} x(t) + \begin{bmatrix} 1 \\ 0 \\ 1 \end{bmatrix} u(t). \tag{3.8}$$

With the proper transformation matrix (2.6), and by scaling the states with damped natural frequency ω_d, the scaled modal form of the system is

$$\dot{\tilde{x}}(t) = \begin{bmatrix} 0 & 0 & 0 \\ 0 & -0.03 & 1 \\ 0 & -1 & -0.03 \end{bmatrix} \tilde{x}(t) + \begin{bmatrix} 1 \\ 0.03 \\ 1 \end{bmatrix} u(t). \tag{3.9}$$

Assuming the final state values $x(t_f) = x_f$ are an equilibrium, the shifted $y(t) = \tilde{x}(t) - x_f$ following (2.14), (2.15) and (2.16) can also be made. So that the state-space equation has become

$$\dot{y}(t) = \begin{bmatrix} 0 & 0 & 0 \\ 0 & -0.03 & 1 \\ 0 & -1 & -0.03 \end{bmatrix} y(t) + \begin{bmatrix} 1 \\ 0.03 \\ 1 \end{bmatrix} u(t). \tag{3.10}$$

Figure 3.1 shows an illustration of the reachable set for (3.10) for a final time t_f of 5 seconds. The sketch only consists of points on the surface of $\mathcal{R}(t)$ which have been generated by using a set of randomly selected η and computing w^* according to the mapping (3.6). The set consists of all points in the state-space from which $y = 0$ can be reached in 5 seconds by using any admissible $u(t)$.

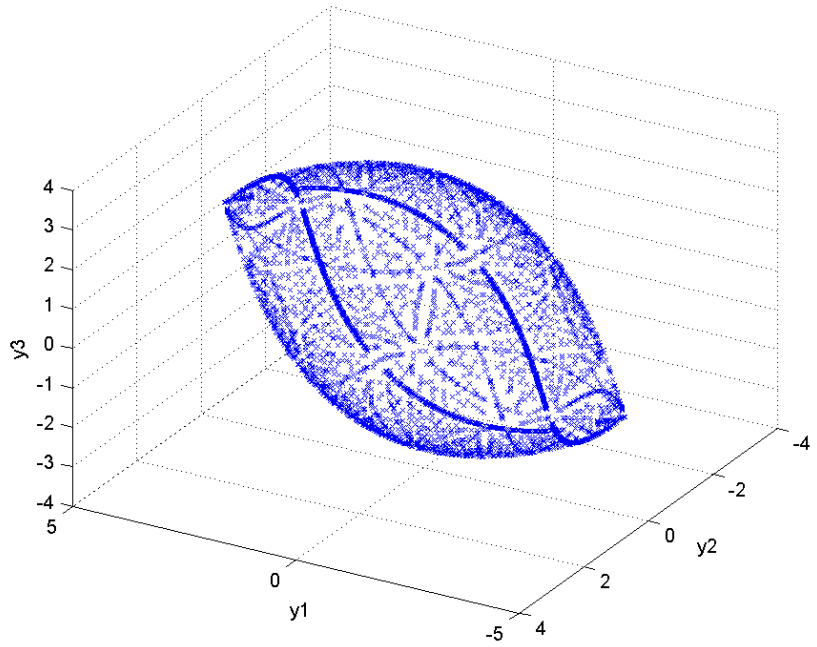


Figure 3.1: Example of 3D reachable set for motion of a flexible structure

Each w^* on the surface represents the point $w^* = [y_1 \;\; y_2 \;\; y_3\,]^T = -\tilde{x}_0$ (assuming $x_f = 0$) and η is an outward normal direction vector at this specific point. For rest-to-rest motion, it is required that x_2 and x_3 at initial and final time are equal to zero (no residual vibration). This corresponds to w^* lying on the first primary axis for y_1 where y_2 and y_3 equal zero.

3.2 Algorithm

For computation of the control solution it will be assumed that the initial state value x_0 and the final state value x_f are known and that x_f corresponds to an equilibrium. With this assumption, the corresponding target point in the surface of the reachable set is given by $w(t) = -x_0$ which may be expressed by $x_0 = \gamma d$ where $-d$ is the required direction for state transfer and γ is the distance to be traveled.

The algorithm proposed here is based on a construction of the reachable set for fixed final time t_f. The algorithm iterates over $\eta \in \mathbb{R}^n$ such that $y(t_f)$ approaches a target point on the boundary of the reachable set given by $w^* = \gamma^* d$. This provides a solution to the fixed final-time optimal control problem defined by:

Fixed final-time optimal control problem. *For given values of t_f and $d \in \mathbb{R}^n$, obtain value of η such that $y(t_f) = \gamma^* d$ for some value of γ^*, where $y(t_f)$ is obtained by the integration* (3.4) *subject to* (3.5).

3.2.1 Procedure

The algorithm is defined by the following steps:

Initialization Set the iteration index: $k = 0$. Determine a set of linearly independent vectors $Y_k = \{y_1, y_2, \ldots, y_n\}$ such that $Y_k \subset \mathcal{R}(t_f)$ and $d \in \text{cone}\,(Y_k)$.

Iteration

1. Compute the coefficients α_i for d as a (positive) combination of the elements of Y_k:

$$d = \sum_{i=1}^{n} \alpha_i y_i, \quad \alpha_i \geq 0, \tag{3.11}$$

2. Compute a lower bound for γ^* as

$$\underline{\gamma}_k = \frac{1}{\sum_i \alpha_i} \tag{3.12}$$

3. Compute the outward normal direction for the hyperplane containing Y_k. i.e. compute a vector n_k satisfying

$$n_k^T y_1 = n_k^T y_2 = \cdots = n_k^T y_n = \kappa \geq 0. \tag{3.13}$$

4. Compute a new point on the surface of the reachable set by using n_k as the initial value of the co-state vector by using mapping (3.6):

$$w_k = f(\eta, t_f), \quad \eta = n_k.$$

5. Compute an upper bound for γ^* as

$$\bar{\gamma}_k = \frac{n_k^T w_k}{n_k^T d}. \tag{3.14}$$

6. If the point w_k satisfies a suitable stopping criterion then stop, otherwise proceed to step 7.

7. Consider the set of cones $j = 1, ..., n$ generated by $\{w_k\} \cup \{y_i\}; i \neq j$ where, for each cone, the element y_j in Y_k is replaced with w_k. Determine which of these cones contains d and use the corresponding set of vectors to form Y_{k+1}. The algorithm then repeats from step 1 with k incremented by 1.

3.2.2 Explanation and remarks

The following gives further explanation and analysis, numbered in correspondence with the steps given previously:

1. As $d \in \text{cone}(Y_k)$ then, by definition, $\alpha_i \geq 0,\ i = 1, 2, \ldots, n$ (see [11]).

2. The point $\underline{\gamma}_k d$ can be expressed as a convex combination of the elements of Y_k:

$$\underline{\gamma}_k d = \frac{1}{\sum_i \alpha_i} \sum_{i=1}^{n} \alpha_i y_i, \quad \alpha_i \geq 0 \tag{3.15}$$

Thus, we have $\underline{\gamma}_k d \in \text{conv}(Y_k) \subset \mathcal{R}(t_f)$ and so (3.12) gives a lower bound for γ^*.

3. Note that n_k is an outward normal to the surface of the reachable set at w_k. As $\mathcal{R}(t_f)$ is convex, we have

$$n_k^T w_k \geq n_k^T x, \quad \forall x \in \mathcal{R}(t_f). \tag{3.16}$$

4. To calculate w_k, a time-optimal u is obtained from (3.5) with $\eta = n_k$. Thus, according to (3.7) switch-times for u_i must be computed as the zeros of $\left[\eta^T X(t) B\right]_i$ over $(0, t_f)$. Herein lies one of the main technical issues for the routine: ensuring

that all the zeros are found, and with sufficient accuracy. With this achieved, w_k can be evaluated by an analytical integration of (3.4) where the state transition matrix $X^{-1}(t)$ is usually expressed as the exponential matrix e^{At}. Thus, from (3.3) and (3.4), suppose that the single optimal control input contains l switches with $u(t) = 1;\ t > t_1$, we then have

$$\begin{aligned} y(t_f) &= \int_0^t X(\tau)Bu(\tau)d\tau \\ &= \sum_{i=0}^{l}(-1)^i \int_{t_i}^{t_{i+1}} X(t)Bdt \end{aligned} \tag{3.17}$$

where $t_0 = 0$ and $t_{l+1} = t_f$. With $\int_{t_i}^{t_{i+1}} X(\tau)d\tau = (-1)^i A^{-1}(e^{-At_{i+1}} - e^{-At_i})$, this series of integrations is reduced to the sum:

$$\begin{aligned} y(t) &= -A^{-1}[(e^{-A(t_1)} - e^{-A(0)}) - (e^{-A(t_2)} - e^{-At_1}) + ... \\ &\quad +(-1)^{l+1}(e^{-A(t_f)} - e^{-At_l})]B \\ &= A^{-1}\left[I + \sum_{i=1}^{l}(-1)^{i+1}2e^{-At_i} + (-1)^i e^{-At_f}\right]B \end{aligned} \tag{3.18}$$

Please note that, for the rigid body mode, A_0 is a singular matrix and thus, A^{-1} does not exist. However, the calculation involving the first and second states (displacement and velocity) can then be done by a simple integration and double integration respectively.

5. Considering w_k as a linear combination of the elements of Y_k:

$$w_k = \sum_{i=1}^{n} \beta_i y_i, \tag{3.19}$$

then

$$\sum_{i=1}^{n} \beta_i = \frac{n_k^T w_k}{n_k^T y_i}. \tag{3.20}$$

Equations (3.16) and (3.20) imply that $\sum_i \beta_i \geq 1$. Also, considering (3.16) with $x = \gamma^* d$ we obtain the upper bound for γ^* stated in step 5:

$$\gamma^* \leq \bar{\gamma}_k = \underline{\gamma}_k \sum_{i=1}^{n} \beta_i = \frac{n_k^T w_k}{n_k^T d}. \tag{3.21}$$

Because of the normal property of system (2.1), $\mathcal{R}(t_f)$ is strictly convex and so w_k is an exposed point on the boundary of $\mathcal{R}(t_f)$. Consequently, $\sum_i \beta_i \to 1$ implies $w_k \to \gamma^* d$.

6. A stopping criterion could take one of various forms according to the type of error that is important. The lower and upper bound for γ^* obtained in each iteration provide a useful indication of convergence. However, it is the error in the system states that may be more important from a practical point of view. It is important here to recognize that an error in $y(t_f)$ corresponds to an error in the initial state x_0. In practice, however, the solution for u is likely to be implemented with the initial state being an exact rest state. Thus, any error in this solution actually leads to residual motion after time t_f caused by a non-zero value for $x(t_f)$. In this case, it is sensible to calculate $x(t_f)$ and use this in the stopping criterion, for example,

$$e = \|x(t_f)\| < \epsilon \tag{3.22}$$

where $x(t_f) = X(t_f)(x_0 + y(t_f))$ with $x_0 = -\gamma d$.

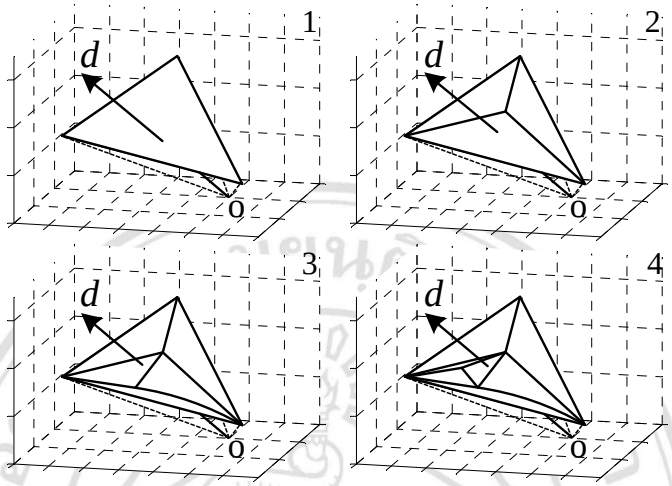
7. The hyperplane containing the elements of Y_k, as defined by (3.13), separates the y-space such that one half-space contains the origin while the other half-space contains w_k and the optimal point $\gamma^* d$. That this is the case follows from the convexity property $n_k^T w_k > n_k^T y_i$ and the property $\gamma^* n_k^T d > \underline{\gamma}_k n_k^T d = n_k^T y_i$. It then follows that including w_k in the basis set Y_{k+1} allows an increase in the lower bound for γ^*. One element of Y_k is also eliminated such that $d \in \text{cone}(Y_{k+1})$ and the algorithm continues by iteration.

3.2.3 Discussion

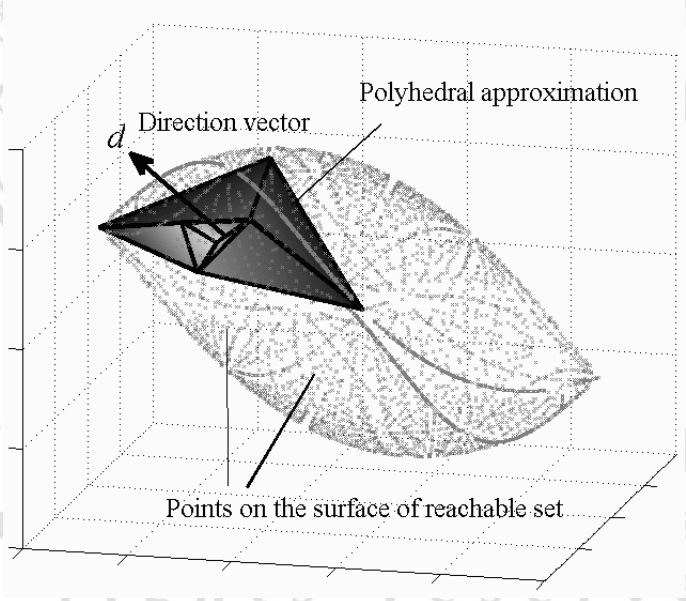
The first issue to be discussed is how to choose elements for the initial basis Y_0. It should be noted that the elements of Y_0 do not necessarily have to be on the boundary of the reachable set and this allows a practical solution which is to generate a set of n linearly independent vectors, denoted Y^0 such that, $d \in \text{cone}(Y^0)$. We can then choose $Y_0 = \kappa Y^0$ with the scalar $\kappa > 0$ sufficiently small to ensure $Y_0 \in \mathcal{R}(t_f)$. As the algorithm progresses, the elements of Y_0 are successively replaced until, at some iteration, all the elements of Y_k are on the surface of $\mathcal{R}(t_f)$.

The overall process can be considered as a construction of the interior of the reachable set by polyhedrons. At each iteration, one facet of the polyhedron is extended by adding n more facets. The approximation becomes more refined as we get closer to the target point in the boundary of the reachable set, as illustrated in Fig. 3.2. The example

here is for a system with three states. For this case, a selection of points on the surface of the reachable set have also been generated by evaluation of $f(\eta, t_f)$ for a complete range of η values. This allows an appreciation of the convex form of the reachable set, which in this case is rather like a rugby ball/football (see Fig. 3.2b).



(a) Refinement of facets at each iteration



(b) Reachable set with polyhedral approximation

Figure 3.2: Illustration of algorithm in $\mathbb{R}^3$. At each iteration one facet is replaced by three more refined facets such that one facet always contains the desired direction of state transfer.

According to step 3, the basis sets Y_k formed at each iteration define a series of hyperplanes that intersect the line extending from the origin in the direction of d. By using a normal to the hyperplane as the initial co-state vector to generate the next point for the

basis, the intersection points move increasingly further from the origin and converge to w^*. For each iteration, the lower bound $\underline{\gamma}_k$ will increase as long as d is contained in the interior of cone(Y_k). It also possible, though less likely, that d is contained in the boundary of cone(Y_k). In this case, $\alpha_i = 0$ for some index i. If the new point w_k replaces y_i for which $\alpha_i = 0$, the lower bound remains the same. However, it can be shown that the upper bound will then always decrease. Thus convergence continues in this manner until the lower bound starts increasing again. In conclusion, the bounds converge such that $\overline{\gamma}/\underline{\gamma} = \sum_i \beta_i \to 1$ and $\underline{\gamma} \to \gamma^*$.

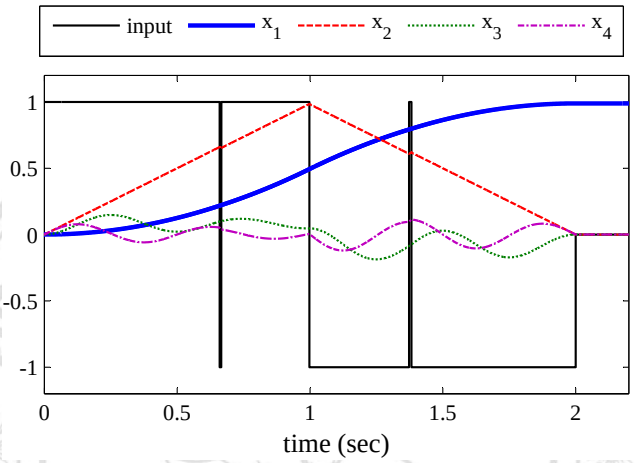
Since the proposed algorithm is based on fixed t_f, to acquire any desired values of γ^*, a simple iteration over t_f can be used, for example, with a bisection algorithm. This guarantees that the optimal control can be found because the reachable set is persistently expanding. Note that the algorithm can also be used directly to calculate a set of optimal control solutions for a range of final time t_f.

3.3 Numerical examples

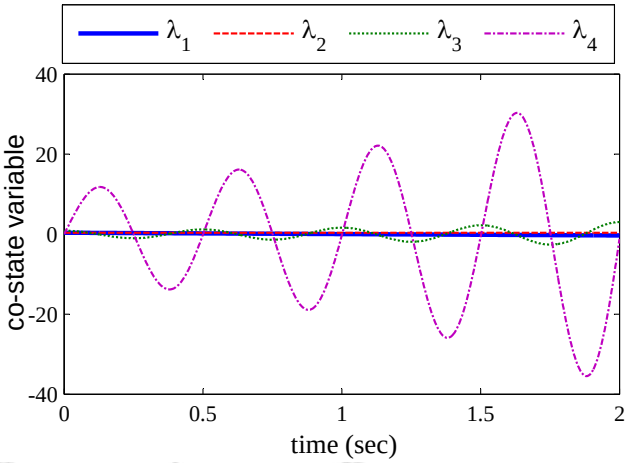
To demonstrate the use of the proposed algorithm, example cases for a single input flexible structure model will be considered. The algorithm is then applied to solve for the time-optimal control input. The calculations for these numerical examples are performed using MATLAB and Simulink toolbox [57]. To find all possible zeros (switching times) as mentioned in the remarks for step 4, the time interval $[0, t_f]$ is discretized into small intervals then the sign of $\left[\eta^T X(t) B\right]$ at each specific time t is inspected. Whenever there is a change of sign between two consecutive inspections, the algorithm applies the MATLAB root-finding command 'fzero' in order to find the switching time t_i where $\left[\eta^T X(t_i) B\right] = 0$. However, there could be a very short interval for a sign change where switches lies between each inspection point and without the successful detection and calculation of all zeroes the algorithm may fail to converge to the desired direction (d). Thus, the discretization interval for t has to be small enough to make sure that all possible switches are detected. This is particularly important as the 'fzero' command only finds one zero within a specified interval.

The model in state space form (2.1) for a flexible structure derived in chapter 2 is considered further in this section. According to (2.2) The states x_1 and x_2 correspond to the position and velocity of the body center-of-mass respectively. The other state vari-

ables are associated with vibratory deformation of the structure with dynamic behavior described by (2.3). The number of vibratory states depends on the number of considered flexible modes. In this section, the overall control task is to achieve a rest-to-rest motion in the sense that no excitation of vibration occurs at either initial or final time. This implies that the desired direction d for state transfer is along the first primary axis: $d = [1,\ 0, \ldots,\ 0]^T$.



(a) Input and state variables



(b) Corresponding co-state variables

Figure 3.3: Optimal Control solution for example case of a single-flexible mode structure

Single-mode case This first example involves the time-optimal motion of a single-mode flexible structure without friction ($c = F_c = 0$). The assigned parameters are $\omega_1 = 8\pi$, $\zeta = 0.05$ and $B_0 = 1$ with $t_f = 2$ sec. The solution shown in Fig. 3.3 was obtained using the algorithm. In most cases the solution for a single-mode system will involve only 3 switches. However, for this example case 5 switches in control input value are required

which shows that the algorithm is capable of finding this more complicated solution. To verify the true optimality of the solution the co-state variable has been calculated following (3.1) by using the final value of η as λ_0. The evolution of $\lambda(t)$ is shown in Fig. 3.3b. Fig. 3.4 shows the value of $\eta^T X(t)B$ whose sign determines u^*.

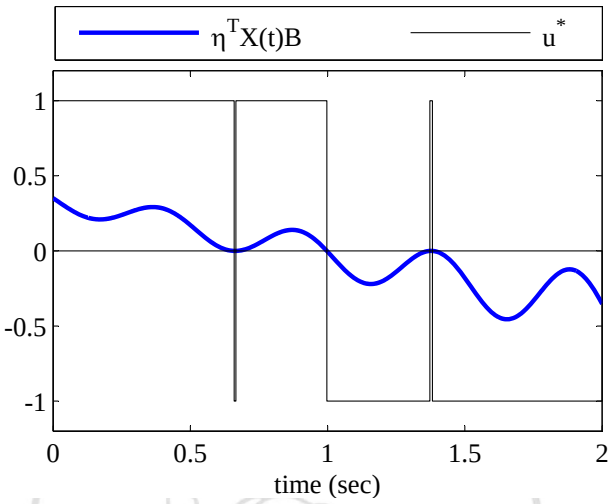


Figure 3.4: Value of $\eta^T X(t)B$ which determines the optimal input u^*

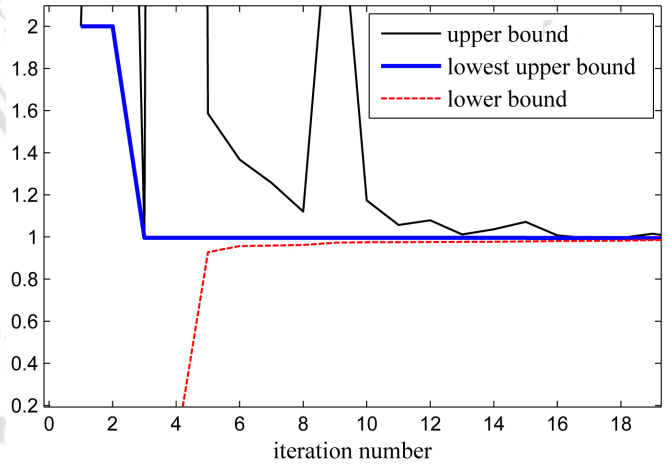
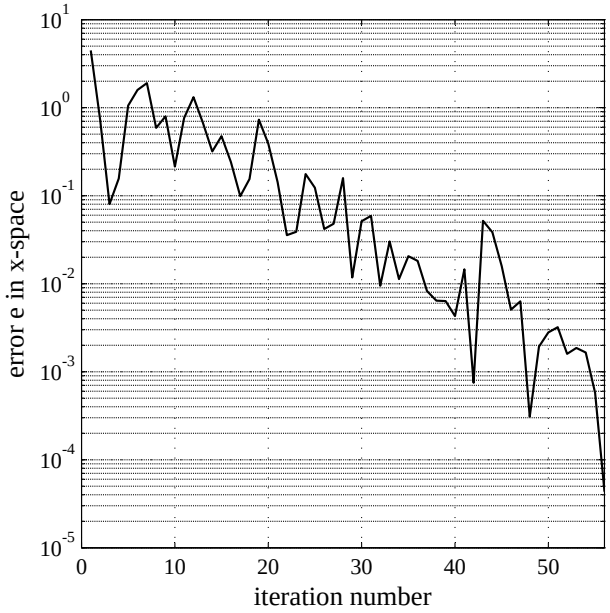


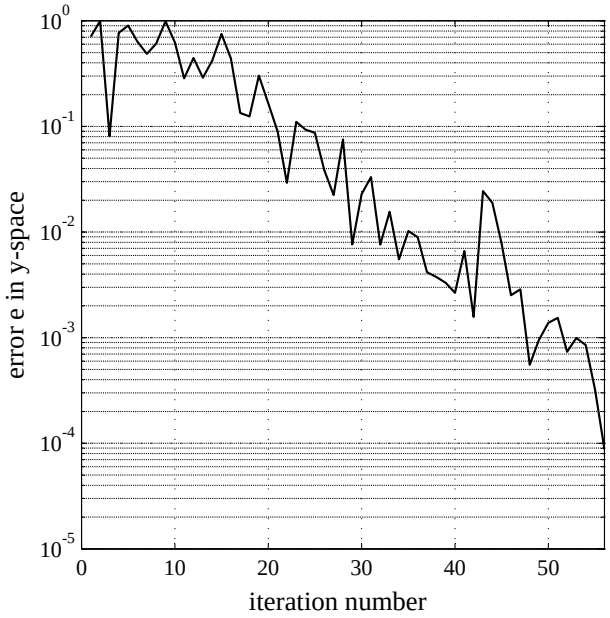
Figure 3.5: Convergence of lower and upper bound for γ^* in example case

The convergence of the lower bound (3.12) and upper bound (3.14) are shown in Fig. 3.5 for this example case. Note that although the lower bound increases every iteration, the upper bound does not. Therefore, Fig. 3.5 also shows the lowest upper bound obtained over all previous iteration. Figure 3.6a shows the measure of error in the solution at each iteration. This measure relates to the error in $x(t_f)$ under application of the solution u and with the nominal initial state (as given in (3.22)). Clearly, it cannot be expected that this

error will decrease every iteration. However, the trend is clearly decreasing and confirms the suitability of using this type of error measure in a stopping criterion. An alternative error measure is the difference between the final point on the surface of reachable set $y(t_f)$ and the desired direction d. This measure is shown in Fig. 3.6b. Physically, it seems to make more sense to use the error in the final value of the state vector since this provides a measure of undesired residual vibration.

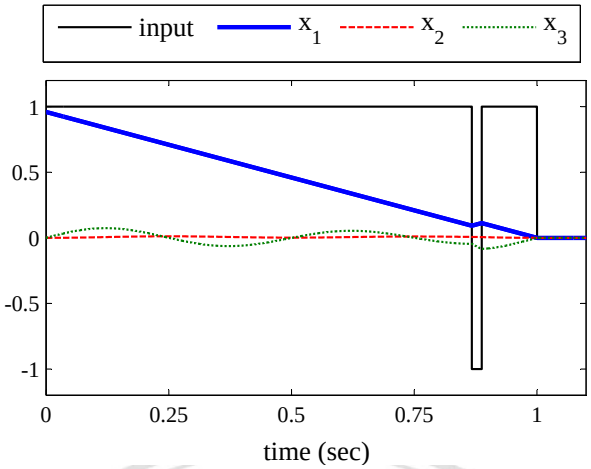


(a) Error in final state $x(t_f)$

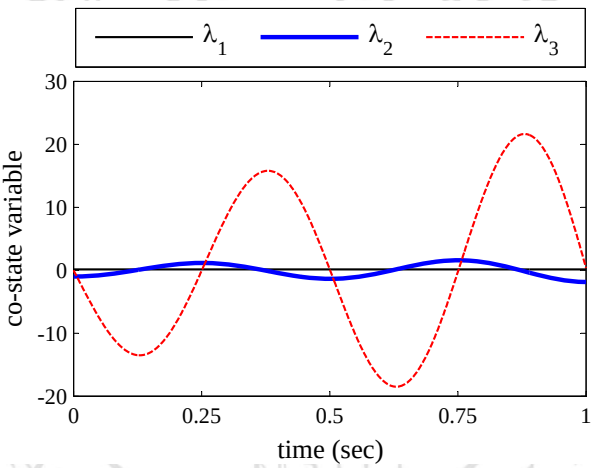


(b) Error in final point on the surface of reachable set $y(t_f)$

Figure 3.6: Error in final state under application of the solution from each iteration



(a) Input and state variables



(b) Corresponding co-state variables

Figure 3.7: Example solution for motion-to-rest task

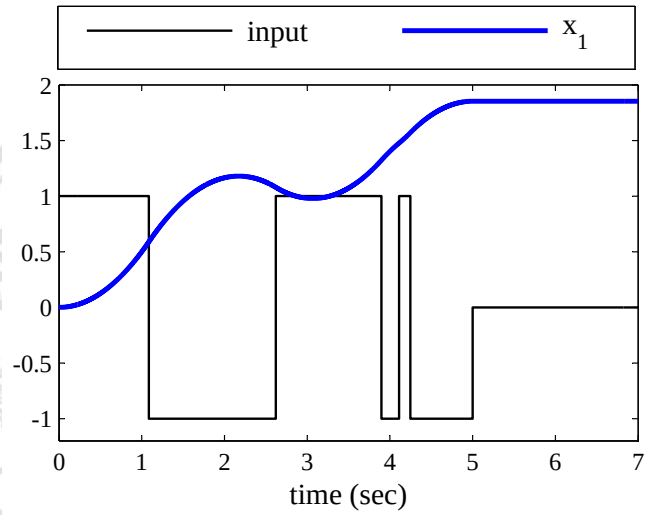
For this single-mode system, an addition case involving a motion-to-rest task has also been considered. The time-optimal objective of this task is to completely stop motion as fast as possible. A similar task would be to accelerate the system from rest to constant speed without vibration in minimum-time (constant residual deflection is allowed). The boundary condition of this situation corresponds to (2.13):

$$x(t_0) = [0\ \nu\ 0\ \ldots\ 0]^T$$

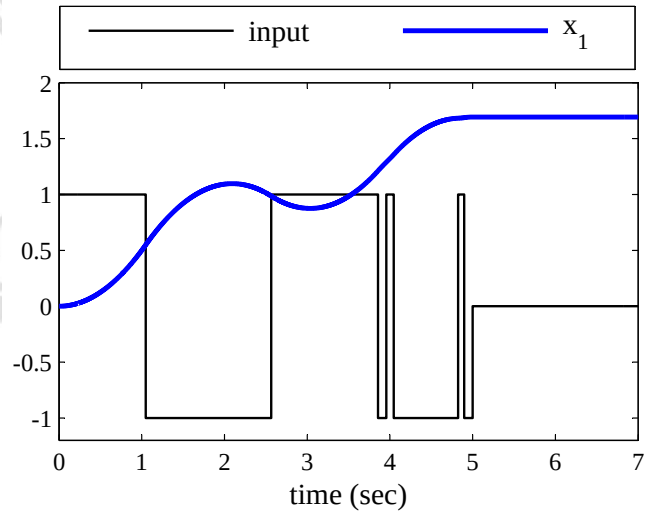
Hence, for this case the first state of the system model (2.2) can be discarded and therefore the first state x_1 represents the overall velocity. This means the total number of states is reduced to three. This 3-state model may also be used for rest-to-rest motions with velocity as a (constrained) input. In this case, the first state represents overall position which is the direct integration of the input. This model may be used to represent a hydraulic servo

system in the real world. The optimal control solution for the non-friction system with $\omega_1 = 4\pi$ rad/sec, $\zeta = 0.05$ and $t_f = 1$ sec together with the corresponding co-state are shown in Fig. 3.7.

Multi-mode cases The general applicability of the proposed algorithm has been confirmed by treating cases involving higher order models with more flexible modes. Example solutions for systems with two and three flexible modes are shown in Fig. 3.8.



(a) 2-flexible-mode model



(b) 3-flexible-mode model

Figure 3.8: Example solution for multi-mode cases

In these cases the natural frequencies are $\omega_1 = 1,\ \omega_2 = 10$ and $\omega_3 = 20$. Damping ratios are $\zeta_i = 0.1, (i = 1, 2, 3)$ and the final time for both cases is $t_f = 5$ sec. The state-space

model of three flexible modes system in this example is:

$$\dot{x}(t) = \begin{bmatrix} 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1^2 & -0.2 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 10^2 & -2 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 & 0 & 20^2 & -4 \end{bmatrix} x(t) + \begin{bmatrix} 0 \\ 1 \\ 0 \\ 1 \\ 0 \\ 1\,0 \\ 1 \end{bmatrix} u(t),$$

$$-1 \leq u(t) \leq 1 \tag{3.23}$$

For clarity, only the first (displacement) state is shown. The number of iterations used to obtain the solution in each case are 154 and 480 respectively. The required number of iterations is generally found to increase with the number of system states (number of flexible modes). This trend relates to the average rate of convergence, as shown in Fig. 3.9 for models of different order. Note, however, that the final time t_f has no obvious influence on the rate of convergence.

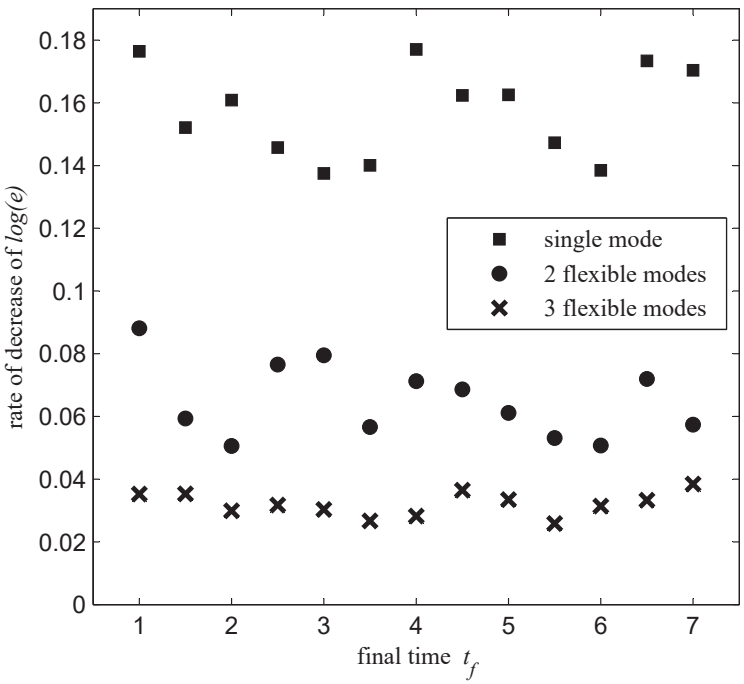


Figure 3.9: Mean rate of decrease of $log(e)$ (per iteration) for a range of cases with different t_f

3.4 Cases with Coulomb friction

The model of mechanical system can be made more general by including friction, since mechanical systems are mostly under the effects of friction forces. In this research Coulomb friction arising between surfaces and acting in the opposite direction to velocity, as described by (2.9), is considered. Including Coulomb friction in the model makes the system dynamics discontinuous with a non-linear term included in the model.

In order to adapt the proposed algorithm for solving cases involving Coulomb friction, it can be noted that if switching times for the friction force $\mathcal{T}_c$ are considered fixed, the reachable set retains the convexity property: the effect of the friction force is simply a fixed translation of $\mathcal{R}(t_f)$. In this case, it is possible to calculate state trajectories using the convex optimization algorithm. The calculated trajectories may then be used to correct the switching times $\mathcal{T}_c$ and this process repeated iteratively until switching times are consistent with the state-trajectories. It is difficult to prove convergence of this process, although it has been shown successful for simple cases. For the cases of motion considered in this section, it is assumed that there is no change in the direction of friction forces during motion, i.e. the direction of motion does not change. With this as an assumption, the convex optimization algorithm can be applied without modification

By letting the system model with Coulomb friction take the form of (2.11), the construction of the reachable set can be done by considering $y(t)$ from (3.4) together with (3.3) and so, the resulting form of $y(t)$ is

$$y(t) = \int_0^t X(\tau)\left(Bu(\tau) + B_c \text{sgn}\left(Cx(t)\right)\right) d\tau \tag{3.24}$$

Assuming that the non-linear signum function does not change value, we have

$$y_F(t) = \pm \int_0^t B_c d\tau \tag{3.25}$$

with the sign depending on motion direction. This can be evaluated by analytical integration similar to (3.18) but does not require the knowledge of the input $u(t)$. Equation (3.25) gives the translational displacement of the reachable set. Figure 3.10 shows the reachable set for rigid body motion where,

$$A = A_0 = \begin{bmatrix} 0 & 1 \\ 0 & -1 \end{bmatrix}, \quad B = B_0 = \begin{bmatrix} 0 \\ 3 \end{bmatrix}, \quad B_c = \begin{bmatrix} 0 \\ \frac{1}{3} \end{bmatrix}$$

and the direction of friction is assumed to be unchanged. The reachable set is shown for both positive (black dot) and negative (blue x) values of friction.

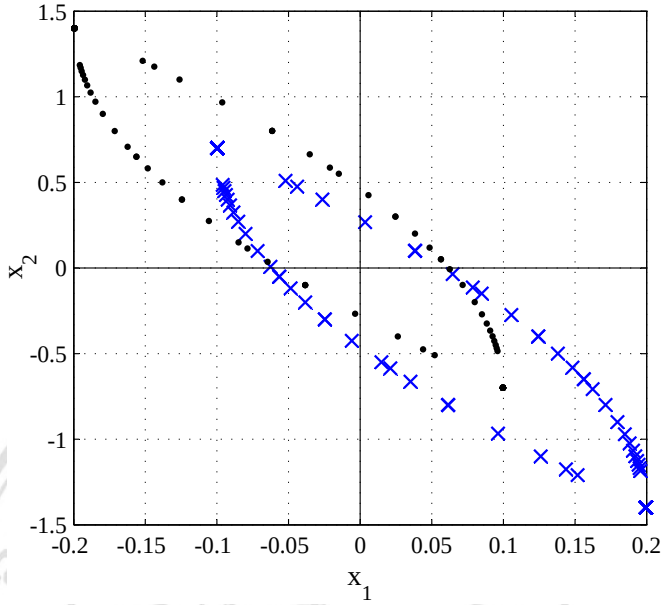


Figure 3.10: Reachable set of rigid body mode with fixed direction of coulomb friction.

Given this constant translation of the reachable set, it is possible that the reachable set will not contain the origin which could cause the convex optimization algorithm to fail. Therefore, in order for the origin to lie within the reachable set, it is required that (3.24) can be made equal to zero with admissible $u(t)$ and hence, it can be implied that

$$Bu(t) + B_c = 0$$
$$B^T Bu(t) + B^T B_c = 0$$
$$u(t) = -(B^T B)^{-1} B^T B_c.$$

$u(t)$, however, must be within the limit of the input. Thus

$$u_i(t) = \left[-(B^T B)^{-1} B^T B_c\right]_i \leq U_i \tag{3.26}$$

This provides a sufficient condition for the origin to be inside the reachable set. Substituting $u(t)$ from (3.26) back into (3.24), it is easy to show that $B(B^T B)^{-1} B^T = I$ for any matrix B with non-singular $B^T B$.

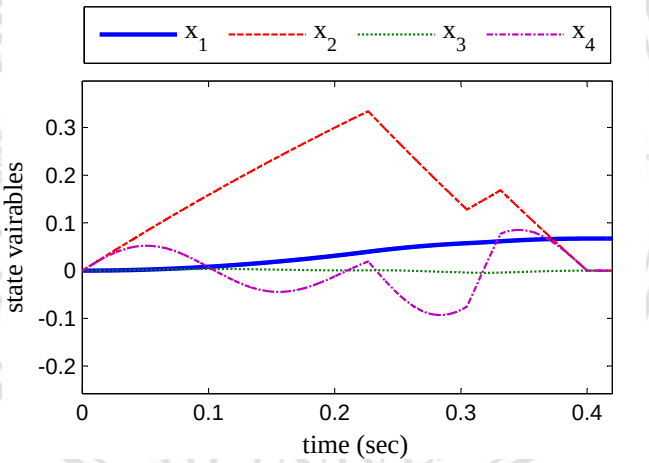
To show that the proposed algorithm can be usefully applied to cases with Coulomb friction, the simulation has been perform based on the parameters for an experimental rig, to be describe in detail in Chapter 6. The simulation results for a solution where the final

time is $t_f = 0.4$ sec and the final distance is 6.7 centimeters are shown in Fig. 3.11. The vibratory state which can be measured by the sensor is x_3. The complete model of this system is:

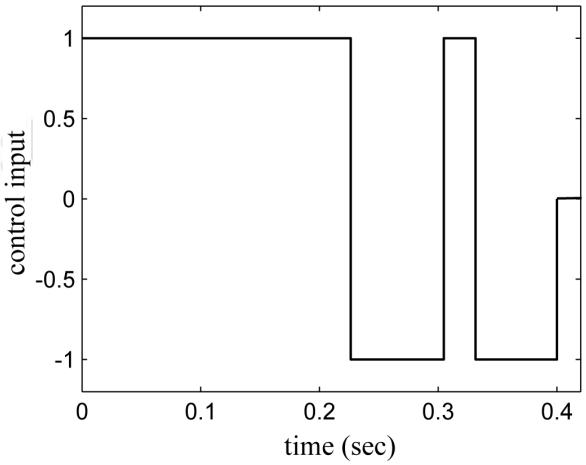
$$\dot{x}(t) = \begin{bmatrix} 0 & 1 & 0 & 0 \\ 0 & -1.21 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & -901.80 & -2.70 \end{bmatrix} x(t) + \begin{bmatrix} 0 \\ 1 \\ 0 \\ 1 \end{bmatrix} u(t) + \begin{bmatrix} 0 \\ -0.34 \\ 0 \\ -0.34 \end{bmatrix} \text{sgn}(x_2) \tag{3.27}$$

and

$$-1 \leq u(t) \leq 1$$



(a) State variables



(b) Time-optimal control solution

Figure 3.11: Simulation results for time-optimal solution based on system with Coulomb friction.