IC-32 **Optimal Control** **Spring 2009**

Benoît Chachuat
ME C2 426, Ph: 35148, benoit@mcmaster.ca

Problem Set #11 (with Solutions)

1. We consider the famous Fuller's problem:

$$\text{minimize:} \quad \mathcal{J}(u) := \int_0^4 [x_1(t)]^2 \, dt \tag{1}$$

$$\text{subject to:} \quad \dot{x}_1(t) = x_2(t); \quad x_1(0) = 2 \tag{2}$$

$$\dot{x}_2(t) = u(t); \quad x_2(0) = -2 \tag{3}$$

$$-1 \le u(t) \le 1, \forall t \in [0, 4]. \tag{4}$$

(a) Using the Pontryagin Maximum Principle, find out the possible types of arc that may be contained in an optimal solution. In particular, show that an optimal solution for the Fuller's problem may have singular arcs of order $p = 2$, and derive the expression u_{sing} of the optimal control along such arcs.

Solution. The Hamiltonian function for the problem (1–4) reads

$$\mathcal{H}(\mathbf{x}, u, \tilde{\boldsymbol{\lambda}}) = \tfrac{1}{2}\lambda_0 x_1^2 + \lambda_1 x_2 + \lambda_2 u.$$

Assuming that $(u^\star, \mathbf{x}^\star, \boldsymbol{\lambda}^\star)$ is an optimal triple for the problem, and that the problem is normal (i.e., $\lambda_0(t) = 1, \forall t$), we have from the Pontryagin Maximum Principle (PMP):

$$u^\star(t) = \begin{cases} 1 & \text{if } \lambda_2^\star(t) < 0 \\ -1 & \text{if } \lambda_2^\star(t) > 0 \\ ? & \text{if } \lambda_2^\star(t) = 0, \end{cases}$$

where

$$\dot{\lambda}_1^\star(t) = -\mathcal{H}_{x_1} = -x_1^\star(t)$$
$$\dot{\lambda}_2^\star(t) = -\mathcal{H}_{x_2} = -\lambda_1^\star(t).$$

That is, singular control arcs are possible when

$$\mathcal{H}_u = \lambda_2^\star(t) = 0,$$

over a finite interval of time. Upon successive differentiation of the foregoing condition with respect to time, we get

$$0 = \frac{d}{dt}\mathcal{H}_u = \dot{\lambda}_2^\star(t) = -\lambda_1^\star(t)$$
$$0 = \frac{d^2}{dt^2}\mathcal{H}_u = -\dot{\lambda}_1^\star(t) = x_1^\star(t)$$
$$0 = \frac{d^3}{dt^3}\mathcal{H}_u = \dot{x}_1^\star(t) = x_2^\star(t)$$
$$0 = \frac{d^4}{dt^4}\mathcal{H}_u = \dot{x}_2^\star(t) = u(t).$$

The problem may therefore have singular arcs of order $p = 2$, and we have

$$u^\star(t) = 0,$$

along such arcs. Moreover, the state and adjoint variables must lie on the singular surface defined by

$$x_1^\star(t) = x_2^\star(t) = \lambda_1^\star(t) = \lambda_2^\star(t) = 0,$$

along a singular arc. Observe also that

$$\frac{\partial}{\partial u}\left[\frac{\mathrm{d}^4}{\mathrm{d}t^4}\mathcal{H}_u\right] = 1 > 0,$$

so that the generalized Legendre-Clebsch condition for a minimum holds along a singular arc.

(b) Although the various types of arcs composing an optimal solution are known, it is hard to find out *a priori* what the optimal arc sequence should be. As a first guess, we shall suppose here that the optimal control sequence consists of 2 arcs,

$$u^{(1)}(t) = \begin{cases} +1 & \text{for } 0 \leq t \leq t_1 \\ u_{\mathsf{sing}} & \text{for } t_1 \leq t \leq 4 \end{cases}$$

In MATLAB®, write a program calculating the optimal values of the junction time t_1 in $u^{(1)}(\cdot)$, so that $\mathcal{J}(u^{(1)})$ is minimized. E.g., this can be done by determining the arc durations Δt_1 and Δt_2 in the following parametric optimization problem:

$$\text{minimize: } x_3(\Delta t_1 + \Delta t_2) \tag{5}$$

$$\text{subject to: } \dot{x}_1(t) = x_2(t); \quad x_1(0) = 2 \tag{6}$$

$$\dot{x}_2(t) = \begin{cases} u^U & \text{for } 0 \leq t \leq \Delta t_1 \\ u_{\mathsf{sing}} & \text{for } \Delta t_1 \leq t \leq \Delta t_1 + \Delta t_2 \end{cases} ; \quad x_2(0) = -2 \tag{7}$$

$$\dot{x}_3(t) = [x_1(t)]^2; \quad x_3(0) = 0 \tag{8}$$

$$\Delta t_1 + \Delta t_2 = 4 \tag{9}$$

$$0 \leq \Delta t_1, \Delta t_2 \leq 4. \tag{10}$$

In particular:

- Use the function `fmincon` to solve the parametric optimization problem (set the solution point tolerance, function tolerance and constraint tolerance to 10^{-10} in `fmincon`);
- Use the function `ode15s` to integrate the differential equations (6–8) on each arc, with state continuity between the arcs (set both the absolute and relative integration tolerances to 10^{-8});
- Consider arcs of equal durations as the initial guess for the optimization;
- For simplicity, let `fmincon` calculate a **finite-difference approximation** of the gradients of the objective function; in particular, set the minimum change in variables for finite differencing to 10^{-4}.

Plot the optimal response vs. time, as well as the optimal response in the phase space (i.e., x_2 vs. x_1). Also get the optimal values of the cost and calculate the function time t_1.

Solution. A generic implementation for any number of arcs is as follows:

FullerMain.m

```
clear all;
clf;
```

```
format long;

% Initial Conditions and Terminal Time
t0 = 0;
tf = 4;
x0 = [2; -2];

% Sequence of arcs
uL = -1;
uU =  1;
uS =  0;
uarc = [ uU; uS ];

% Initial Values and Bounds for the Junction Times
dtarc0 = ones(length(uarc),1)*tf/length(uarc);
dtarcL = ones(length(uarc),1)*0.01;
dtarcU = ones(length(uarc),1)*tf;

% Linear Equality Constraint
dteq   = ones(1,length(uarc));

% Options for SQP & ODE solvers
optSQP = optimset( 'Display', 'iter', 'GradObj', 'off', 'GradConstr', 'on', ...
                   'DerivativeCheck', 'on', 'LargeScale', 'off', ...
                   'HessUpdate', 'bfgs', 'Diagnostics', 'on',  'TolX', 1e-10, ...
                   'TolFun', 1e-10, 'TolCon', 1e-10, 'MaxFunEval', 2000, ...
                   'MaxIter', 1000, 'DiffMinChange', 1e-4 )
optODE = odeset( 'RelTol', 1e-8, 'AbsTol', 1e-8 );

% Optimize Junction Times
[dtopt, fopt, iout] = fmincon( @(dtarc)FullerFunc(dtarc,uarc,x0,t0,tf,optODE), ...
                               dtarc0, [], [], dteq, tf, dtarcL, dtarcU, [], optSQP )
```

FullerFunc.m

```
function [ f ] = FullerFunc( dtarc, uarc, x0, t0, tf, optODE )
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% FullerFunc - This function calculates the value of the objective function
% in the Fuller's problem:
%
%  [ f ] = FullerFunc( dtarc, uarc, x0, t0, tf, optODE )
%
%   dtarc   duration of each arc
%   uarc    (constant) values of the control on each arc
%   x0      initial state (x01, x02)
%   t0      initial time
%   tf      terminal time
%   optODE  options for ODEs integration
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
  % Initial State and Time for First Arc
  tini = t0;
  xini = [ x0; 0 ];
  tstore = tini;
  ystore = xini';
  dtarc;
```

```

  % Loop over all Arcs
  for ijunc = 1:length(uarc);
    % Integrate State and Dynamic Cost Equations on Arc #ijunc
    tfin = tini + dtarc(ijunc);
    if( tfin - tini >= 1e-10 )
      [t,y] = ode15s( @(t,y)FullerDiff(t,y,uarc(ijunc)), [tini tfin], xini, optODE );
      xini = y(length(y),:)';
      % Store State and in Arc #ijunc
      tstore = [tstore; t];
      ystore = [ystore; y];
    end
    tini = tfin;
  end

  % Display States, t0<=t<=tf
  figure(1);
  plot(tstore,ystore(:,1),'-r',tstore,ystore(:,2),'-b');
  figure(2);
  plot(ystore(:,1),ystore(:,2),'-r');

  % Calculate Objective Function
  f = xini(3);
end
```

FullerDiff.m

```
function [ dy ] = FullerDiff( t, y, u )
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% Fullerdiff - This function calculates the right hand side of the differential
% equations in the Fuller's problem:
%
%  [ dy ] = FullerDiff( t, y, u )
%
%   t       current time
%   y       current state & dynamic cost value [y1(t),y2(t),c(t)]
%   u       (constant) value of the control
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
  dy = [ y(2); u; y(1)^2 ];
end
```

The optimal arc durations are found to be $\Delta t_1^\star = \Delta t_2^\star = 2$, i.e., the optimal junction time is $t_1^\star = 2$; the corresponding optimal cost value is $J(u^{(1)}) = 1.6$. A plot of the responses $x_1^\star$ and $x_2^\star$ vs. time is shown in the left plot of Fig. 1. A representation of the response in the phase space is given in the right plot of Fig. 1.

(c) Repeat the foregoing optimization approach for the following arc sequences:

$$u^{(3)}(t) = \begin{cases} -1 & \text{for } 0 \le t \le t_1 \\ +1 & \text{for } t_1 \le t \le t_2 \\ -1 & \text{for } t_2 \le t \le t_3 \\ u_{\mathsf{sing}} & \text{for } t_3 \le t \le 4, \end{cases} \qquad \text{and} \qquad u^{(4)}(t) = \begin{cases} -1 & \text{for } 0 \le t \le t_1 \\ +1 & \text{for } t_1 \le t \le t_2 \\ -1 & \text{for } t_2 \le t \le t_3 \\ +1 & \text{for } t_3 \le t \le t_4 \\ u_{\mathsf{sing}} & \text{for } t_4 \le t \le 4, \end{cases}$$

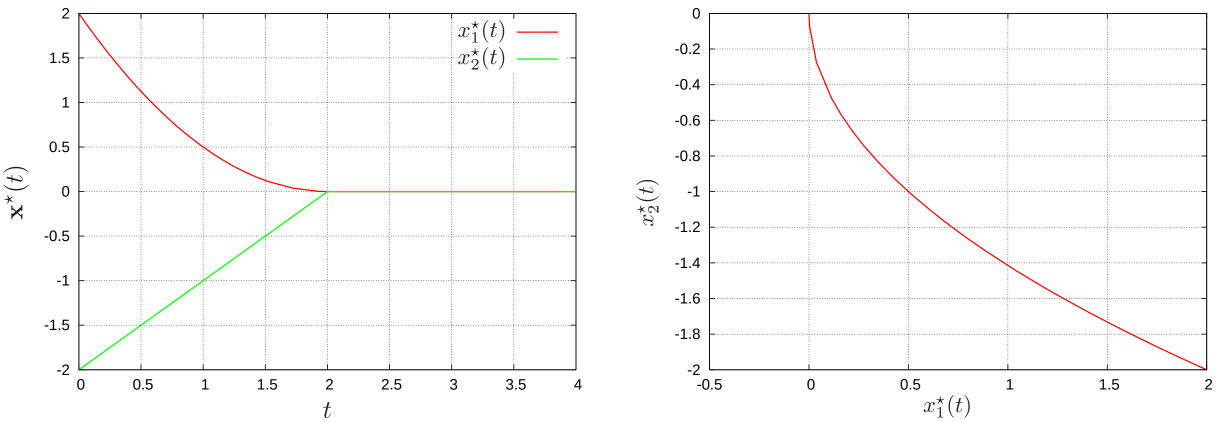

Figure 1: Optimal results for control sequence $u^{(1)}$). Left plot: response $x_1^\star$, $x_2^\star$ vs. t; right plot: response in the phase space.

For each arc sequence, calculate the optimal cost value and junction times, and plot the optimal response vs. time, as well as the optimal response in the phase space. Comment the results.

Solution. Using the previous generic implementation, only the main file requires minor modifications. E.g., for the arc sequence $u^{(4)}(t)$, we have:

FullerMain.m

```
clear all;
clf;
format long;

% Initial Conditions and Terminal Time
t0 = 0;
tf = 4;
x0 = [2; -2];

% Sequence of arcs
uL = -1;
uU =  1;
uS =  0;
uarc = [ uL; uU; uL; uU; uS ];     % <-- ONLY MODIFICATION!

% Initial Values and Bounds for the Junction Times
dtarc0 = ones(length(uarc),1)*tf/length(uarc);
dtarcL = ones(length(uarc),1)*0.01;
dtarcU = ones(length(uarc),1)*tf;

% Linear Equality Constraint
dteq   = ones(1,length(uarc));

% Options for SQP & ODE solvers
optSQP = optimset( 'Display', 'iter', 'GradObj', 'off', 'GradConstr', 'on', ...
                   'DerivativeCheck', 'on', 'LargeScale', 'off', ...
                   'HessUpdate', 'bfgs', 'Diagnostics', 'on',  'TolX', 1e-10, ...
                   'TolFun', 1e-10, 'TolCon', 1e-10, 'MaxFunEval', 2000, ...
                   'MaxIter', 1000, 'DiffMinChange', 1e-4 )
```

```
optODE = odeset( 'RelTol', 1e-8, 'AbsTol', 1e-8 );

% Optimize Junction Times
[dtopt, fopt, iout] = fmincon( @(dtarc)FullerFunc(dtarc,uarc,x0,t0,tf,optODE), ...
                               dtarc0, [], [], dteq, tf, dtarcL, dtarcU, [], optSQP )
```

The optimal junction times and cost values are as follows:

$$u^{(3)}(t) = \begin{cases} -1 & \text{for } 0 \leq t \leq 0.05765 \\ +1 & \text{for } 0.05765 \leq t \leq 2.60918 \\ -1 & \text{for } 2.60918 \leq t \leq 3.11980 \\ u_{\text{sing}} & \text{for } 3.11980 \leq t \leq 4, \end{cases} \qquad \mathcal{J}(u^{(3)}) \approx 1.51525,$$

$$u^{(4)}(t) = \begin{cases} -1 & \text{for } 0 \leq t \leq 0.05766 \\ +1 & \text{for } 0.05766 \leq t \leq 2.61190 \\ -1 & \text{for } 2.61190 \leq t \leq 3.21436 \\ +1 & \text{for } 3.21436 \leq t \leq 3.31684 \\ u_{\text{sing}} & \text{for } 3.31684 \leq t \leq 4, \end{cases} \qquad \mathcal{J}(u^{(3)}) \approx 1.51523$$

The plots of the responses $x_1^\star$ and $x_2^\star$ vs. time, and of the response in the phase space are given in Fig. 2 and 3 for $u^{(3)}(t)$ and $u^{(4)}(t)$, respectively.

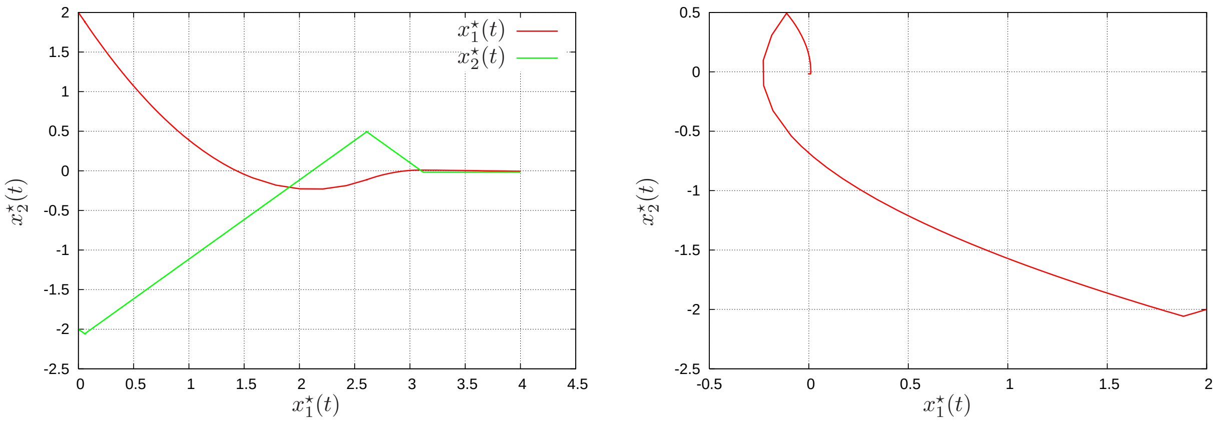

Figure 2: Optimal results for control sequence $u^{(3)}$). Left plot: response $x_1^\star$, $x_2^\star$ vs. t; right plot: response in the phase space.

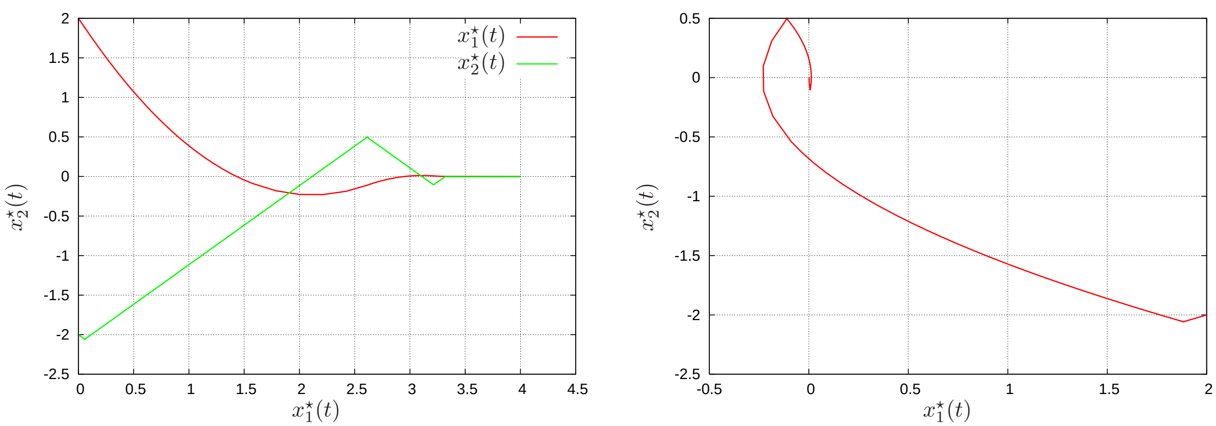

Figure 3: Optimal results for control sequence $u^{(4)}$). Left plot: response $x_1^\star$, $x_2^\star$ vs. t; right plot: response in the phase space.