Optimal Control - Homework Exercise 3

November 1, 2023

In this homework exercise two different problems will be considered, first the so called Zermelo problem where the problem is to steer a boat in streaming water, and then a problem where the thrust angle is controlled to obtain the maximum orbit radius of a space shuttle. The task is to apply different numerical methods to solve these optimal control problems. The two problems are similar in the sense that an angle is the control variable in both problems, but in practise the latter problem is more difficult to treat since it includes constraints on the final states.

Discretization, gradient and shooting methods will be used, as well as the CasADi toolbox. You must write the files of the Zermelo problem by yourself, but for the second problem all needed files can be downloaded, but some lines of code are missing (indicated by three question marks ???). Even though much code is available to save time, it is important that you understand the overall algorithms so that you will be able to solve optimal control problems from scratch if you have to.

Present your results in a brief report (pdf or hand-written) where answers to the exercises and your conclusions, plots, etc., are included. The report (pdf or hand-written) and all m-files must be compressed (zip or tar.bz2) and emailed to the teaching assistant.

1 The Zermelo Problem

A ship is traveling with constant speed w, with respect to the water, through a region where the velocity of the water current is parallel to the x-axis and varies with y. The dynamic ship model can be expressed as

$$\begin{pmatrix} \dot{x}(t) \\ \dot{y}(t) \end{pmatrix} = F(X(t), \theta(t)) = \begin{pmatrix} w\cos(\theta) + v(y) \\ w\sin(\theta) \end{pmatrix} \tag{1}$$

where $X = (x \;\; y)^T$ is the position of the ship and θ is the heading angle of the ship relative to the x-axis.

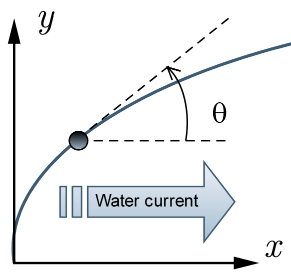

Figure 1: The Zermelo problem in Section 1. Maximize the travel distance in the x-direction by steering the ship. The speed of the water current is dependent on the distance from the shore at $y = 0$.

This problem is a maximum range problem where we want to maximize the travel distance in the x direction. Assume that the initial position of the ship is

$X_0 = (0\ \ 0)^T$ and that the water current is linear with respect to y, i.e. $v(y) = y$. Then, the problem can be formulated as

$$
\begin{aligned}
\min_{\theta(t),\ 0\le t\le t_f} \quad & J = -x(t_f) \\
s.t.\ & \dot{x}(t) = w\cos(\theta(t)) + y(t) \\
& \dot{y}(t) = w\sin(\theta(t)) \\
& x(0) = 0 \\
& y(0) = 0
\end{aligned}
\tag{2}
$$

where t_f is fixed and the optimal solution is

$$\theta(t) = \arctan(t_f - t). \tag{3}$$

1.1 Analytical solution

(a) Write down the Hamiltonian H and calculate and solve the adjoint equations.

(b) Compute $\frac{\partial H}{\partial \theta}$, the partial derivative of the Hamiltonian w.r.t. θ.

(c) Show that (3) is a minimal solution to the problem (2).

(d) Plot the optimal trajectory by using an ODE (e.g. `ode23`) solver in Matlab. Use $t_f = 1$ and $w = 1$. What is the numerical optimal objective value?

1.2 Discretization method solution

We will now use a discretization method to solve the problem in Matlab, read Section 7.5.3 in the book [1]. You must write the main script and the functions by yourself, but exercise 7.11 in the book [1] can be used as a template.

Define the optimization parameter vector as

$$Y = \begin{pmatrix} X[0]^T\ \theta[0] & X[1]^T\ \theta[1]\ \dots\ X[N-1]^T\ \theta[N-1] & X[N]^T\ \theta[N] \end{pmatrix}^T. \tag{4}$$

and the optimization problem is expressed as the constrained problem

$$\min_{y} \quad \mathcal{F}_0(Y) \tag{5}$$

$$s.t. \quad \mathcal{H}(Y) = 0. \tag{6}$$

where

$$\mathcal{F}_0(Y) = -x(t_f) \tag{7}$$

and

$$\mathcal{H}(Y) = \begin{pmatrix} h_1(Y) \\ h_2(Y) \\ h_3(Y) \\ \vdots \\ h_{N+1}(Y) \end{pmatrix} = \begin{pmatrix} X[0] - X_i \\ X[1] - \bar{F}(X[0], \theta[0]) \\ X[2] - \bar{F}(X[1], \theta[1]) \\ \vdots \\ X[N] - \bar{F}(X[N-1], \theta[N-1]) \end{pmatrix} \tag{8}$$

(a) Let T denote the sample time and use $t_f = 1$. Derive a discrete-time model of (1)

$$X[k+1] = \bar{F}(X[k], \theta[k]), \tag{9}$$

where

$$v(y) = y, \quad w = 1$$

and $X[k] = X(kT)$, by using the Euler-approximation.

(b) Write a script and functions that solves the discrete-time version of the problem with the Matlab function `fmincon`. Use the discrete-time model derived in (a) to define the non-linear constraints and define a discrete-time version of the objective function. Also compute the gradients of the constraint and the objective function, respectively. The gradient is the second output argument of the functions. To tell the optimization function to use the gradient set the optimization parameters `GradObj` and `GradConstr` to `'on'`, see below.

```
options = optimset('fmincon');
options = optimset(options, 'Algorithm','interior-point');
options = optimset(options, 'GradObj','on');
options = optimset(options, 'GradConstr','on');
options = optimset(options, 'MaxFunEvals',15000);
[X,fval] = fmincon(@zermeloCostCon, X0, [], [], ...
  Aeq, Beq, [], [], @zermeloNonlcon, options, T);
```

(c) Compares the solution to the optimal solution in Section 1.1. What is the numerical value of the cost?

(d) Set `GradObj` and `GradConstr` to `'off'` and run the script. Make comments about the result and the computation time and explain the difference.

1.3 Gradient method

A gradient search approach will next be used to solve the problem. Read Section 7.5.1 in the book [1]. Use the files in exercise 7.13 in the book [1] as templates.

(a) Solve the problem by using a gradient search approach based on the time-continuous model in (1), the adjoint equations from 1.1 (a), and the gradient expression from 1.1 (b). As above, use $t_f = 1$. *Hint: You might need to increase the stepsize α for the algorithm to converge*

(b) What is the numerical value of the cost? Compare the solution to the solutions in Sections 1.1 and 1.2. Which method do you prefer, the discretization or gradient method?

2 Max radius orbit transfer in a given time

Consider a rocket engine with constant thrust operating for a given time t_f. The problem is to find the thrust-direction $\theta(t),\ 0 \le t \le t_f$, to transfer a shuttle from a given initial radius orbit to the largest possible orbit. Make the following definitions:

- T; constant thrust force.
- $\bar{r}$; radial distance from the attracting center (the sun).
- $\bar{u}$,$\bar{v}$; radial and tangential components of velocity, respectively.
- m; mass of the shuttle, the fuel consumption rate $-\dot{m}$ is constant.
- θ; thrust direction angle.
- μ; gravitational constant of attracting center.

To simplify the presentation the following dimensionless variables are defined

$$r(t) = \frac{1}{\bar{r}(0)}\bar{r}(t), \quad u(t) = \frac{\bar{r}^2(0)}{\mu t_f}\bar{u}(t), \quad v(t) = \sqrt{\frac{\bar{r}(0)}{\mu}}\bar{v}(t) \tag{10}$$

and for a particular choice of the constant parameters the problem can be expressed as

$$\begin{aligned} \min_{\theta(t),\ 0\le t\le t_f} \quad & J = -r(t_f) \\ s.t.\ & \dot{x} = F(x,\theta) \\ & x(0) = x_0 \\ & x(t_f) \in S_f \end{aligned} \tag{11}$$

where state vector is $x = (r \ \ u \ \ v)^T$ and the dynamic model is

$$\dot{x} = \begin{pmatrix} \dot{r} \\ \dot{u} \\ \dot{v} \end{pmatrix} = \begin{pmatrix} u \\ \frac{v^2}{r} - \frac{1}{r^2} + a\sin\theta \\ -\frac{uv}{r} + a\cos\theta \end{pmatrix} \tag{12}$$

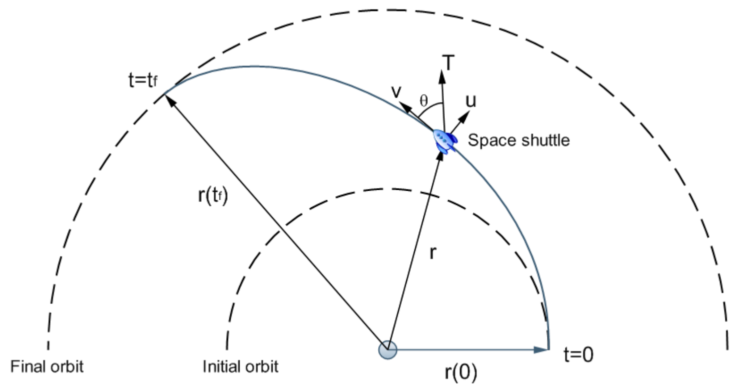

Figure 2: The problem in Section 2. Maximize the orbit radius by controlling the direction of the thrust.

and $a(t) = T/(m_0 - |\dot{m}|t)$, $m_0 = 1$. The initial radius is 1 and the initial velocity is 1 in the tangential direction. Thus, the initial state is $x_0 = (1 \;\; 0 \;\; 1)^T$. The terminal constraints of u and v are derived from the conditions that the radial speed must be 0 and the centrifugal force and gravitational force must be balanced, i.e.,

$$S_f = \{x \in \mathbb{R}^3 | G(x) = 0\}. \tag{13}$$

where

$$G(x) = \begin{pmatrix} u \\ v - \frac{1}{\sqrt{r}} \end{pmatrix}. \tag{14}$$

2.1 Boundary condition iteration (Shooting)

To solve this optimal control problem a TPBV problem will be defined and solved by applying a boundary condition iteration (shooting) algorithm. Read Section 7.5.2 in the book [1]. The idea here is to take small steps to the desired solution, i.e., that the terminal constraints are zero. The Matlab function `fsolve` will be used to update the initial guess of $\lambda(0)$. Note that the shooting approach can handle terminal state constraints easier than the basic gradient algorithm used in Section (1.3), but the drawback is that a suitable start guess of $\lambda(0)$ is needed.

(a) Define $\lambda = (\lambda_r \;\; \lambda_u \;\; \lambda_v)^T$ and write down the Hamiltonian and derive the adjoint equations.

(b) The terminal constraints of the adjoint equations are given by

$$\lambda(t_f) = \lambda_0 \nabla\phi(x(t_f)) + G_x(x(t_f))^T \nu, \tag{15}$$

for some $\nu \in \mathbb{R}^2$ (see Section 7.4 in the book [1]). We will only consider the "normal" case here where $\lambda_0 = 1$.

There are three terminal constraints in total. Two state constraints are given by (13) and the last one is

$$\lambda_r(t_f) + 1 - \frac{\lambda_v(t_f)}{2[r(t_f)]^{3/2}} = 0. \tag{16}$$

Derive this constraint based on (15).

(c) Show that the control signal θ can be expressed as a function of λ, i.e., derive

$$\tan\theta = \frac{\lambda_u}{\lambda_v}. \tag{17}$$

(d) The optimal control problem can be formulated as a TPBV problem defined by the system equations in (12), the adjoint equation that is derived in (a), the inital state condition x_0 and the terminal conditions in (b). All the needed files already exist, you should only replace the three questions marks with suitable code.

Complete the function `maxRadiusOrbitTransferEqAndAdjointEq.m` with the system model (12) and the adjoint equations. Note that they are stacked to be able to be solved concurrently. Then write the terminal constraints in the function `terminalStateCondition.m`.

Run the main script `mainMaxRadiusOrbitTransferShooting.m`, but note that you have to find an suitable initial value of $\lambda(0)$. What is the numerical value of the optimal cost? Which start guess of $\lambda(0)$ did you use? What happens if a bad initial $\lambda(0)$ is used? *Hint: The final radius should be around 1.5.* $\lambda(0) = (-1 \quad -1 \quad -1)^T$ *should be ok as a start guess.*

2.2 CasADi

Finally, the CasADi toolbox will be used to solve the optimal control problem.

Complete the function `mainMaxRadiusOrbitTransferCasADi.m` with the system model (12), the terminal constraints (13), and the cost function. Solve the problem with CasADi (see the initializing instructions in the exercises of chapter 7 [1]). What is the numerical value of the optimal cost?

References

[1] Hansson A. and Andersen M. Optimization for learning and control. John Wiley & Sons, 2023.