AA274A: Principles of Robot Autonomy I
Course Notes

Oct 4, 2019

5 Advanced methods for trajectory optimization

Last lecture, we looked at a specific case of deriving a closed-loop control law for differential flat systems using Lyapunov. In this lecture, we will look into a more general case of solving the trajectory planning problem as an optimal control problem with two boundary conditions. Instead of limiting ourselves to differentially flat systems, we will use indirect methods to derive an open-loop control law for the optimal control problem by enforcing necessary conditions for optimality (NOC).

We start by first reviewing constrained optimization, where the optimization is over vectors of finite dimension. We'll then see how this extends to optimization over infinite dimensional vectors, i.e. functions like $\boldsymbol{u}(t)$ which we'll use to solve optimal control problems.

5.1 Constrained Optimization

Much of this section is a direct excerpt from [BV04].
Let's consider an optimization problem in the standard form:

$$\begin{aligned} \min_{u} \quad & f(x) \\ \text{subject to} \quad & f_i(x) \leq 0, \quad i = 1, \dots, m \\ & h_i(x) = 0, \quad i = 1, \dots, p \end{aligned} \tag{1}$$

with variable $x \in \mathbb{R}^n$.

The key idea for solving this constrained optimization problem is to use the Lagrangian, which takes the constraints in (1) into account by augmenting the objective function with a weighted sum of the constraint functions. We define the *Lagrangian* $L : \mathbb{R}^n \times \mathbb{R}^m \times \mathbb{R}^p \longrightarrow \mathbb{R}$ associated with the (1) as

$$L(x, \lambda, \nu) = f_0(x) + \sum_{i=1}^{m} \lambda_i f_i(x) + \sum_{i=1}^{p} \nu_i h_i(x).$$

We refer to λ_i as the *Lagrange multiplier* associated with the ith inequality constraint $f_i(x) \leq 0$; similarly, we refer to ν_i as the Lagrange multiplier associated with the ith equality constraint $h_i(x) = 0$. In the context of AA274A, we will only handle the equality constraints, and we will use λ to indicate the Lagrange multiplier for all our constraints.

It can be shown that the greatest lower bound of $L(x, \lambda)$ yields the lower bounds on the optimal value of (1). Therefore, our necessary conditions for optimality are x^* and λ^* that sets partial derivatives of $L(x, \lambda)$ to 0, or:

$$\begin{aligned} \nabla_x L(x^*, \lambda^*) = 0 \\ \nabla_\lambda L(x^*, \lambda^*) = 0. \end{aligned} \tag{2}$$

5.2 Indirect methods and necessary conditions for optimality

Much of this section is a direct excerpt from [Kir04].

In this chapter we first derive necessary conditions for optimal control assuming that the admissible controls are not bounded. These necessary conditions are then employed to find the optimal control law for the important linear regulator problem.

As discussed in lecture 3, our goal is to find an admissible control u^* that causes the system

$$\dot{\boldsymbol{x}} = \boldsymbol{a}(\boldsymbol{x}(t)\boldsymbol{u}(t), t)$$

to follow an admissible trajectory x* that minimizes the performance measure

$$J(\boldsymbol{u}) = h(\boldsymbol{x}(t_f), t_f) + \int_{t_0}^{t_f} g(\boldsymbol{x}(t), \boldsymbol{u}(t), t)dt. \tag{3}$$

We shall initially assume that the admissible state and control regions are not bounded, and that the initial conditions $x(t_0) = x_0$ and the initial time t_0 are specified. As usual, $x \in \mathbb{R}^n$ is the state vector and $u \in \mathbb{R}^m$ is vector of control inputs. Assuming that h is a differentiable function, we can write

$$h(\boldsymbol{x}(t_f), tf) = \int_{t_0}^{t_f} \frac{d}{dt}[h(\boldsymbol{x}(t), t)]dt + h(\boldsymbol{x}(t_0), t_0), \tag{4}$$

so that the performance measure can be expressed as

$$J(\boldsymbol{u}) = \int_{t_0}^{t_f} \big\{g(\boldsymbol{x}(t), \boldsymbol{u}(t), t) + \frac{d}{dt}[h(\boldsymbol{x}(t), t)]\big\}dt + h(\boldsymbol{x}(t_0), t_0). \tag{5}$$

Since $x(t_0)$ and $t_0)$ are fixed, the minimization does not affect the $h(\boldsymbol{x}(t_0), t_0)$ term, so we need consider only the functional

$$J(\boldsymbol{u}) = \int_{t_0}^{t_f} \big\{g(\boldsymbol{x}(t), \boldsymbol{u}(t), t) + \frac{d}{dt}[h(\boldsymbol{x}(t), t)]\big\}dt. \tag{6}$$

To include the differential equation constraints, we form the augmented functional

$$
\begin{aligned}
J(\boldsymbol{u}) = \int_{t_0}^{t_f} \big\{ & g(\boldsymbol{x}(t), \boldsymbol{u}(t), t) \\
& + \big[\frac{\partial h}{\partial \boldsymbol{x}}(\boldsymbol{x}(t), t)\big]^T \dot{\boldsymbol{x}}(t) \\
& + \frac{\partial h}{\partial t}(\boldsymbol{x}(t), t) \\
& + \boldsymbol{p}^T(t)[\boldsymbol{a}\boldsymbol{x}(t), \boldsymbol{u}(t), t) - \dot{\boldsymbol{x}}(t)]\big\} dt.
\end{aligned} \tag{7}
$$

by introducing the Lagrange multipliers, or *costates*, $p_1(t), \ldots p_n(t)$.

We can now take a partial derivative of J with respect to $\boldsymbol{x}, \dot{\boldsymbol{x}}, \boldsymbol{u}, \boldsymbol{p}$ to determine the variation of J_a. Detailed derivation is skipped for the sake of space, but we direct interested readers to [Kir04]. The results of this derivation state that $\boldsymbol{p}$ and $\boldsymbol{x}$ that meet the following conditions sets $\partial J = 0$:

$$
\begin{aligned}
\dot{\boldsymbol{x}}^*(t) &= \frac{\partial H}{\partial \boldsymbol{p}}(\boldsymbol{x}^*(t), \boldsymbol{u}^*(t), \boldsymbol{p}^*(t), t) \\
\dot{\boldsymbol{p}}^*(t) &= -\frac{\partial H}{\partial \boldsymbol{x}}(\boldsymbol{x}^*(t), \boldsymbol{u}^*(t), \boldsymbol{p}^*(t), t) \\
0 &= \frac{\partial H}{\partial \boldsymbol{u}}(\boldsymbol{x}^*(t), \boldsymbol{u}^*(t), \boldsymbol{p}^*(t), t)
\end{aligned} \tag{8}
$$

and the boundary conditions are necessary to set the integral term to be 0:

$$
\begin{aligned}
& [\frac{\partial h}{\partial \boldsymbol{x}}(\boldsymbol{x}^*(t_f), t_f) - \boldsymbol{p}^*(t_f))]^T \delta \boldsymbol{x}_f \\
& + [H(\boldsymbol{x}^*(t_f), \boldsymbol{u}^*(t_f), \boldsymbol{p}^*(t_f), t_f) + \frac{\partial h}{\partial t}(\boldsymbol{x}^*(t_f), t_f)]\delta t_f = 0
\end{aligned} \tag{9}
$$

where the Hamiltonian H is defined as

$$
H := g(\boldsymbol{x}(t), \boldsymbol{u}(t), t) + \boldsymbol{p}^T(t)[a(\boldsymbol{x}(t), \boldsymbol{u}(t), t]. \tag{10}
$$

The conditions in Eq. (8) are the necessary conditions for optimality (also referred to as necessary optimality conditions, NOCs). Notice that these necessary conditions consist of a set of $2n$, first-order differential equations (the state and costate equations), and a set of m algebraic relations which must be satisfied throughout the interval $[t_0, t_f]$). The solution of the state and costate equations will contain $2n$ constants of integration. To evaluate these constants we use then equations $x^*(t_0) = x_0$ and an additional set of n or $(n-1)$ relationships—depending on whether or not t_f is specified—from Eq. 9.

5.3 Boundary conditions

Determining the boundary conditions is a matter of making the appropriate substitutions in Eq. 9. In all cases it will be assumed that we have the n equations $\boldsymbol{x} * (t_0) = \boldsymbol{x}_0$.

Problems with Fixed Final Time. If the final time r, is specified, $x(t_f)$ may be specified, free, or required to lie on some surface in the state space.

CASE I. *Final state specified.* Since $\boldsymbol{x}(t_f)$ and t_f are specified, we substitute $\delta\boldsymbol{x}_f$ and $\delta t_f = 0$ in (9). The required n equations are

$$\boldsymbol{x}^*(t_f) = \boldsymbol{x}_f. \tag{11}$$

CASE II. *Final state free.* Since $\boldsymbol{x}(t_f)$ is arbitrary, the n equations

$$\frac{\partial h}{\partial \boldsymbol{x}}(\boldsymbol{x}^*(t_f), t_f) - \boldsymbol{p}^*(t_f) = 0 \tag{12}$$

must be satisfied.

Problems with Free Final Time. If the final time is free, there are againn two situations that may occur.

CASE I. *Final state specified.* The appropriate substitution in Eq. 9 is $delta\boldsymbol{x}_f = 0$. δt_f is arbitrary, so the $(2n+1)$st relationship is:

$$[H(\boldsymbol{x}^*(t_f), \boldsymbol{u}^*(t_f), \boldsymbol{p}^*(t_f), t_f + \frac{\partial h}{\partial t}(\boldsymbol{x}^*(t_f), t_f)]\delta\boldsymbol{x}_f = 0. \tag{13}$$

CASE II. *Final state free.* $\delta\boldsymbol{x}(t_f)$ and δt_f are arbitrary and independent; therefore, their coefficients must be zero. this means that we have n equations from

$$\boldsymbol{p}^*(t_f) = \frac{\partial h}{\partial \boldsymbol{x}}(\boldsymbol{x}^*(t_f)) \tag{14}$$

and one equation from

$$[H(\boldsymbol{x}^*(t_f), \boldsymbol{u}^*(t_f), \boldsymbol{p}^*(t_f), t_f + \frac{\partial h}{\partial t}(\boldsymbol{x}^*(t_f), t_f)]\delta\boldsymbol{x}_f = 0 \tag{15}$$

that must be satisfied.

5.4 Extension of boundary conditions

Frequently, Boundary Value Problems (BVP) have boundary conditions (BC) which are not in the "standard" form that BVP solvers can solve. For example, a BVP may involve BC at an unknown point, or have a free final time. In these cases, it is possible to add trivial

ODEs to correspond to the unknown values and transform the interval to one with known, fixed endpoints to turn the problem into a "standard" BVP [AR81].

As discussed in [AR81], the procedure to transform the unknown time interval into a variable in the ODE is following:

1. Rescale time $t \longrightarrow \tau$, where $\tau = \frac{t}{t_f}$, and $\tau \in [0, 1]$.
2. Change derivatives using chain rule: $\frac{d}{d\tau} := t_f \frac{d}{dt}$.
3. Introduce dummy state r that corresponds to t_f with dynamics $\dot{r} = 0$.
4. Replace all instances of t_f with r.

We look at a specific example to describe how to apply indirect methods and reformulation of BVP in the standard form.

5.5 Solving optimal control problem with free final time as a 2P-BVP problem

The last scenario is what we call a free final time state and introduces an unknown variable to our system of equations making it a set of 2N + M + 1. Think of this as an optimal control problem where we look to minimize the cost of our function at any given final time. We look at a thorough example for clarification.

Consider a robot that moves in a 1-dimensional space, x. Our control input is acceleration of the robot. We're given the boundary conditions

$$\ddot{x} = u, \quad x(0) = 10, \quad \dot{x} = 0, \quad x(t_f) = 0, \quad \dot{x}(t_f) = 0 \tag{16}$$

but the final time is free. We measure the performance of the robot with the cost function

$$J = \frac{1}{2}\alpha t_f^2 + \frac{1}{2}\int_{t_0}^{t_f} bu^2(t)\mathrm{d}t, \tag{17}$$

which means our terminal cost function is $h(x, u, t) = \frac{1}{2}\alpha t^2$, and our path cost function is $g(x, u, t) = \frac{1}{2}bu^2(t)$. Note the quadratic cost functions shown here are very common in control problems. $h(x, u, t)$ tries to minimize total time to get back to origin, while $g(x, u, t)$ attempts to keep acceleration from being too high, leading to excessive control effort. Note that a and b are 'weights' in our cost function, J. Larger b means that our cost function will penalize large control effort harshly. An optimal solution found with this cost function will result in minimal control effort, i.e., low acceleration and unacceptably long t_f. On the other hand, larger α suggests our optimal control solution will penalize larger final time harshly, i.e., our control law will be "encouraged" to move faster.

Now we're ready to solve the problem. We first start by building our state vector $\mathbf{z}$. To handle the second derivative $\ddot{x}$, we can define new variables x_1, x_2:

$$\begin{aligned} \dot{x_1} &= x_2 \\ \dot{x_2} &= u, \end{aligned} \tag{18}$$

and Eq. (18) will work as equality constraints on our system for our state vector $\boldsymbol{z}$. Since we get a costate variable for every constraint in the system, we will need to add p_1 and p_2 in our state vector. Lastly, we add a dummy state r in our state space to solve for the final time.

Combined, we arrive at state vector $\boldsymbol{z}$:

$$\mathbf{z} = [z_1, z_2, z_3, z_4, z_5]^T = [x_1^*, x_2^*, p_1^*, p_2^*, r^*]^T \tag{19}$$

Next, we derive Hamiltonian. We already know that $h(x, u, t) = \frac{1}{2}\alpha t^2$, and $g(x, u, t) = \frac{1}{2}bu^2(t)$, which leads us to

$$H = \frac{1}{2}bu^2 + p_1 x_2 + p_2 u. \tag{20}$$

With the state vector $\boldsymbol{z}$ and Hamiltonian, we're ready to derive the necessary conditions of optimality:

$$\begin{aligned}
\dot{x}_1^* &= \frac{\partial H}{\partial p_1}(\mathbf{x}^*(t), u^*(t), \mathbf{p}^*(t), t) = x_2^* \\
\dot{x}_2^* &= \frac{\partial H}{\partial p_2}(\mathbf{x}^*(t), u^*(t), \mathbf{p}^*(t), t) = u^* \\
\dot{p}_1^* &= -\frac{\partial H}{\partial x_1}(\mathbf{x}^*(t), u^*(t), \mathbf{p}^*(t), t) = 0 \\
\dot{p}_2^* &= -\frac{\partial H}{\partial x_2}(\mathbf{x}^*(t), u^*(t), \mathbf{p}^*(t), t) = -p_1^* \\
0 &= \frac{\partial H}{\partial u}(\mathbf{x}^*(t), u^*(t), \mathbf{p}^*(t), t) = bu^* + p_2^*
\end{aligned} \tag{21}$$

Using the last equation in 21, we can rewrite u^* as:

$$u^* = -\frac{1}{b}p_2^*. \tag{22}$$

Next, we impose boundary conditions. This flavor of boundary condition problem has free final time t_f, and a fixed final state $\boldsymbol{x}(t_f)$. Therefore, our boundary conditions can be reduced to

$$H(\mathbf{x}^*(t_f), u^*(t_f), \mathbf{p}^*(t_f), t_f) + \frac{\partial h}{\partial t}(\mathbf{x}^*(t_f), t_f) = 0, \tag{23}$$

which, by substitution, is

$$\frac{1}{2}bu^*(t_f)^2 + p_1^*(t_f)x_2^*(t_f) + p_2^*(t_f)u^*(t_f) + \alpha t_f = 0. \tag{24}$$

Using the boundary conditions

$$x_1^*(0) = 10, x_2^*(0) = 0, x_1^*(t_f) = 0, x_2^*(t_f) = 0, \tag{25}$$

and relationships obtained from (21), our boundary conditions reduced to

$$-\frac{1}{2b}p_2^*(t_f)^2 + \alpha t_f = 0. \tag{26}$$

Now we rewrite our necessary conditions of optimality in (21). After rescaling time with $\tau = t/t_f$, our NOC equations become

$$\frac{\mathrm{d}\mathbf{z}}{\mathrm{d}\tau} = t_f \frac{\mathrm{d}\mathbf{z}}{\mathrm{d}t} = z_5 \begin{bmatrix} 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & -\frac{1}{b} & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & -1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \end{bmatrix} \mathbf{z}. \tag{27}$$

Also using the boundary condition values, we obtain

$$\begin{gathered} z_1(0) = 10, z_2(0) = 0, z_1(1) = 0, z_2(1) = 0, \\ -\frac{1}{2b}z_4(1)^2 + \alpha z_5(1) = 0. \end{gathered} \tag{28}$$

Now we have finished casting the optimal control problem into 2V-BVP. Our job is done here – numerical solution can now be found using open source tools, such as `scikits.bvp_solver` . Note in this case, it is also possible to solve for $z_5 = t_f$ analytically, and obtain the solution

$$t_f = (1800b/\alpha)^{1/5}. \tag{29}$$

The derivation is left as an exercise for the reader.

References

[AR81] Uri Ascher and Robert D Russell. Reformulation of boundary value problems into "standard" form. *SIAM review*, 23(2):238–254, 1981.

[BV04] Stephen Boyd and Lieven Vandenberghe. *Convex optimization*. Cambridge university press, 2004.

[Kir04] Donald E. Kirk. *Optimal Control Theory: An Introduction (Dover Books on Electrical Engineering)*. Dover Publications, 2004.